# Choice-driven dial-a-ride problem for demand responsive mobility service

Sharif Azadeh, Sh; Atasoy, Bilge; Ben-Akiva, Moshe E.; Bierlaire, M.; Maknoon, M. Y.

**DOI**
10.1016/j.trb.2022.04.008

**Publication date**
2022

**Document Version**
Final published version

**Published in**
Transportation Research Part B: Methodological

**Citation (APA)**
Sharif Azadeh, S., Atasoy, B., Ben-Akiva, M. E., Bierlaire, M., & Maknoon, M. Y. (2022). Choice-driven dial-a-ride problem for demand responsive mobility service. *Transportation Research Part B: Methodological*, *161*, 128-149. https://doi.org/10.1016/j.trb.2022.04.008

Transportation Research Part B 161 (2022) 128–149

Contents lists available at ScienceDirect

Transportation Research Part B

journal homepage: www.elsevier.com/locate/trb

# Choice-driven dial-a-ride problem for demand responsive mobility service

Sh. Sharif Azadeh [a,c,d,*], Bilge Atasoy [b,c], Moshe E. Ben-Akiva [c], M. Bierlaire [d], M.Y. Maknoon [e,c,d]

[a] Department of Transport & Planning, Delft University of Technology, Netherlands
[b] Department of Maritime and Transport Technology, Delft University of Technology, Netherlands
[c] Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, USA
[d] School of Architecture, Civil and Environmental Engineering, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[e] Faculty of Technology, Policy, and Management, Delft University of Technology, Netherlands

ARTICLE INFO

*Keywords:*
Choice-driven dial-a-ride problem
Mixed-Integer Linear Problem (MILP)
Assortment optimization
Demand responsive mobility

ABSTRACT

Urban mobility services face the challenge of planning their operations efficiently while complying with user preferences. In this paper, we introduce a new mathematical model called a *choice-driven dial-a-ride problem* (CD-DARP) which is a generalization of the dynamic DARP where passenger behavior is integrated in the operational planning using choice models and assortment optimization. We look at two types of mobility services, private and shared. Our problem extends the dynamic DARP by (i) changing its objective function to profit maximization, where both cost and revenue are variables, and (ii) incorporating assortment optimization with routing decisions in a dynamic setting. We propose a pricing scheme based on a choice model designed to offer service alternatives at the time a customer makes a request. We introduce a tailored algorithm to efficiently solve the dynamic CD-DARP. Computational results indicate that our proposed approach outperforms dynamic DARP in terms of reducing routing costs and improving the number of customers served.

## 1. Introduction

The *dial-a-ride problem* (DARP) is a variation of the *pickup and delivery problem* (PDP) involving passenger transportation systems. A solution of DARP requires balancing the trade-off between service quality (i.e. customer convenience) and economic perspective, Paquette et al. (2009). In the case of the dial-a-ride problem, service quality is either handled externally (e.g. setting a tight time window to restrict customer waiting time and limit the maximum ride time to reduce any inconvenience caused by detour) or included internally as a part of the objective function, Cordeau and Laporte (2007).

In the dynamic case, new requests arrive continuously in the system. When a new request is submitted, vehicle routes need to be adjusted which requires efficient strategies for dispatching vehicles (e.g. wait or go) and routing. The routing decision involves either rejecting the customer when there is no feasible solution or accepting the request and guaranteeing the service at the lowest routing cost, Berbeglia et al. (2010).

In literature, there are two main approaches to finding a balance between service quality and operational cost. In the first approach, customer inconvenience (due to early or late arrival) is modeled as a soft constraint whose violation is penalized in the

* Corresponding author at: Department of Transport & Planning, Delft University of Technology, Netherlands.
*E-mail addresses:* s.sharifazadeh@tudelft.nl (Sh. Sharif Azadeh), b.atasoy@tudelft.nl (B. Atasoy), mba@mit.edu (M.E. Ben-Akiva), michel.Bierlaire@epfl.ch (M. Bierlaire), M.Y.Maknoon@tudelft.nl (M.Y. Maknoon).

https://doi.org/10.1016/j.trb.2022.04.008
Received 14 April 2021; Received in revised form 19 April 2022; Accepted 29 April 2022
Available online 21 May 2022

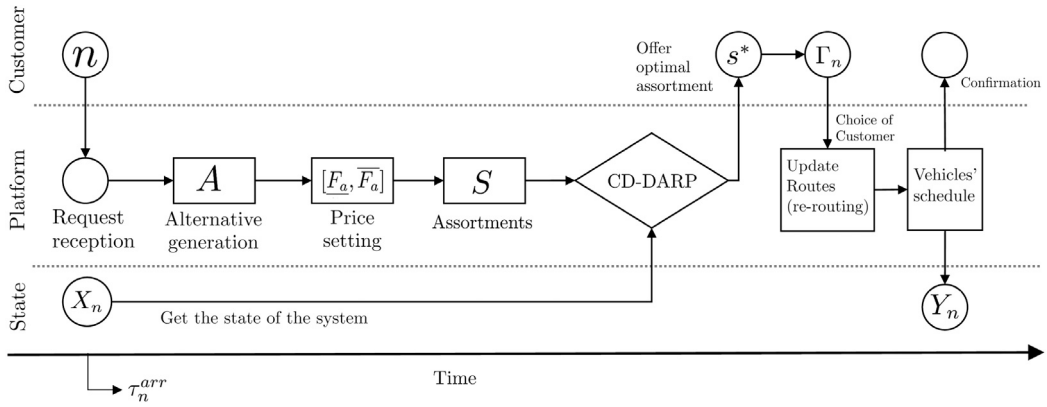



**Fig. 1.** Timeline of the dynamic framework.

objective function (see, Jorgensen et al. (2007) and Melachrinoudis et al. (2007)). In the second approach, service quality is viewed as part of the objective function. For example, Lehuédé et al. (2014) use weighted-sum method, while Schilde et al. (2011, 2014) apply hierarchical objective functions. Parragh et al. (2009), Paquette et al. (2013), and Molenbruch et al. (2017b) use Pareto dominance method to tackle the problem. Defining a model that incorporates the interaction between the minimization of operational costs and the maximization of service levels is not straight-forward. This is especially critical for demand-responsive mobility services dealing with heterogeneous customers in a competitive market, Atasoy et al. (2015). In Section 2, we present some of the recently published researches that address this integration.

DARP literature largely assumes that the users of a system do not have a voice in the service provision beyond formulating transportation requests. They are usually not assumed to have any kind of choice to make. The underlying context is that of a public service that aimed at serving as much of the demand as possible under a budget constraint or to serve all demand at minimal cost.

Inspired by the utility maximization theory of consumer behavior and discrete choice models, initially presented by Ben-Akiva et al. (1985) and McFadden (1973), in this paper, we propose a variant of dynamic dial-a-ride problem called dynamic *choice-driven dial-a-ride problem* (CD-DARP). Upon customers' arrival in the system, we generate a set of options called *personalized alternatives* that matches the preferences of the new request in terms of the type of service (e.g., private or shared) and the preferred pickup time. Via assortment optimization, a menu of *personalized alternatives* is offered to the customer at different *price levels* to maximize profit at the network level. In this paper, we solve the problem as the requests arrive in the system in real-time. Fig. 1 in Section 3 clarifies the details of our proposed approach.

The fundamental trade-off in assortment optimization is that broad assortments result in demand cannibalization and spoilage, while narrow assortments result in disappointed customers who may opt-out without purchasing. An assortment's profitability can be captured by a choice model that provides the probability of purchase based on the alternatives offered in the assortment. While integrating individual preferences makes the dynamic DARP more realistic, that comes with greater computational complexity due to the probabilistic representation of demand, Paneque et al. (2021). In our case, choice parameters are estimated outside the optimization problem.

We summarize the contributions of this paper as follows: we introduce a novel mathematical model that incorporates customer choice in the dynamic DARP framework. Personalized alternatives are generated based on customer's preferences. We investigate the structural properties of inherited assortment model to obtain a set of assortments. These properties are used to efficiently solve the model. We then present an algorithm to solve the dynamic CD-DARP. We test our approach on synthetic as well as New York taxi data. We also compare our approach with the dynamic dial-a-ride. Our results indicate that, by providing flexible services, we can serve higher number of customers while reducing the routing cost and increasing the profit.

The remainder of the paper is structured as follows. Relevant literature is discussed in Section 2. In Section 3, we present the dynamics of the system. Sections 4 and 5 describe the elements of CD-DARP and its mathematical formulation. Section 6 explains the variable reduction techniques. To tackle the problem dynamically, we use the rolling horizon approach presented in Section 7. In Section 8, we show the numerical results and finally our findings are summarized in Section 9.

## 2. Related literature

The main application of dial-a-ride problem arises in door-to-door transportation services offered to elderly and handicapped people in many cities. For instance, both Madsen et al. (1995) and Toth and Vigo (1996) investigate DARP for ambulance and emergency services. Beaudry et al. (2010) consider the case for urban mobility systems such as demand responsive transport and shared autonomous vehicles, see also, Bongiovanni et al. (2019), Levin (2017), Parragh et al. (2015) and Marković et al. (2015). In this section, we limit our review to the dynamic DARP research. Interested readers are referred to Ho et al. (2018), Berbeglia et al.

(2010, 2007), Molenbruch et al. (2017a) and Cordeau and Laporte (2007) for an extensive and comprehensive review on dial-a-ride problems.

In the literature of dynamic DARP, customer convenience has been modeled implicitly by defining for example, tight time windows and restricted maximum ride time for each new request. In this case, the decision-making process focuses on finding a feasible low-cost solution to serve the incoming request. Berbeglia et al. (2011) present a constraint programming approach to efficiently detect the feasibility of serving a new request. The proposed method was then combined by a tabu search to find a low-cost solution, Berbeglia et al. (2012). Similarly, Attanasio et al. (2004) propose a heuristic algorithm to efficiently accommodate the incoming requests (see also, Horn (2002), Coslovich et al. (2006) and Xiang et al. (2008), Liu et al. (2015)).

Service quality has been treated as part of the solution and quantified directly in the objective function. With this approach, the operator can balance the service quality and economic benefits manually. For a single-vehicle dynamic DARP, Psaraftis (1980) presents a dynamic programming approach designed to minimize a weighted combination of user dissatisfaction and total service time, while Häme (2011) extends the work of Psaraftis (1980) and makes it computationally more efficient. Beaudry et al. (2010) developed an adaptive insertion algorithm that takes waiting time and fleet size into account. For the transportation of elderly people, Madsen et al. (1995) propose a heuristic approach by taking into account factors like driving time, vehicle utilization and routing cost. For the cases mentioned above, operational decisions are quantified mostly by routing cost, vehicle utilization and the number of served customers.

Knowing future arrivals, several studies use the queuing model to determine a social-optimum solution by finding a balance between service quality and operating costs. For last-mile transportation systems, Wang and Odoni (2016) propose a queuing model that schedules vehicle plans to minimize weighted sum of passenger waiting time, in-vehicle time and vehicle workload. For online DARP, Hyytiä et al. (2012) relax the constraints related to customer convenience (i.e. pickup/delivery time windows and maximum ride time) and propose a queuing model to estimate the future cost while not allowing for vehicle re-routing.

Service pricing is a commonly used approach to offer personalized services. Customer satisfaction perceived value is a frequently used approach to measure service attractiveness. It is mostly defined as a difference between the gained utility by using the service (e.g. waiting time, in-vehicle time) expressed as a generalized cost and a dis-utility associated with price. Customers with a positive perceived value will join the system, see, Santos and Xavier (2015) and Qian et al. (2017) for taxi service and Huang et al. (2020) for demand-responsive buses. Taking future arrivals into account, Sayarshad and Gao (2018) extend the queuing method proposed by Hyytiä et al. (2012) using the concept of customer perceived value and proposing an approach designed to determine a social optimum pricing solution for on-demand systems, see also, Sayarshad and Chow (2015).

Despite their potential to measure satisfaction, customer perceived value cannot predict the behavior of individuals. Discrete choice models, on the other hand, are widely used to measure satisfaction and predict customer behavior for a variety of transportation problems, see, Dias et al. (2017) and Zhao et al. (2018). For on-demand systems, Liu et al. (2019) study a mode choice problem to design an on-demand system where travel mode demand is a function of its service level. Krueger et al. (2016) conduct a targeted survey on the preferences and adaptation of shared and private autonomous vehicles. Karamanis et al. (2018) embed discrete choice models within an agent-based framework to simulate the impact of utility-based pricing for shared and private services, see also, Qiu et al. (2018). The recent research trend to use discrete choice models in transport planning to capture people's behavior shows their potential to be integrated in the dial-a-ride problems. This way, operators can make more customer-friendly decisions. In recent years, similar methods have also been adapted for the attended home delivery problems. Although, the context of city logistics and urban mobility are different in nature, still there is a great deal of similarities in the way these problems can be formulated, in terms of delivery time windows,offered assortments, and routing. Interested readers are referred to the following recently published research on the topic, Köhler et al. (2020), Bruck et al. (2018), Mackert (2019), Ulmer and Thomas (2020).

The body of literature on assortment optimization is quite extensive. We refer the reader to Kök et al. (2008) for an overview of assortment optimization problems using discrete choice models. Assortment optimization has been investigated for several parametric choice models, including *multinomial logit* model (MNL) (see also, Wang (2012), Talluri and van Ryzin (2004)) and *nested-logit* (NL) (Davis et al., 2014).

On-demand mobility systems (a special application of dynamic DARP) are usually characterized by the presence of three conflicting objectives: maximizing the number of served customers, minimizing routing costs and maximizing user satisfaction. In this paper, we present a model with the aim of maximizing profit. In Sections 8.3–8.5 we use *personalized alternatives*, *assortment and trip-based pricing,* to show how to reach these objectives. In most assortment optimization problems, the costs associated with alternatives being offered is supposed to be known so, revenue and profit maximization are considered to be equivalent which is not the case for routing problems, as the costs of each alternative depend on the inherited routing cost.

In this paper, we present an optimization approach that simultaneously determines the optimal vehicle routes and the assortment being offered to the customer. We take individual behavior into account using assortment optimization. In Section 3, we explain in detail the dynamic framework of this problem.

## 3. Dynamics of the system

We consider a privately owned on-demand operator for an urban mobility system who is committed to providing private and shared on-demand services. Each vehicle can change its service type throughout a day. For the shared mobility service, several trips can be served up until the maximum capacity of the vehicle has been reached.

The operator receives $N$ trip requests at time $\tau_1^{arr} < \tau_2^{arr} < \cdots < \tau_N^{arr}$, where $N$ is a random variable. The requests arrive one at a time (i.e., batch arrivals are excluded) and are processed sequentially, based on the first-come, first-served principle. The operator

has no prior knowledge of future demand (such as arrival rate and origins–destinations) and any demand surge does not influence the price of alternatives. The aim is to identify the most profitable set of alternatives to offer to each new request to maximize the expected profits by solving a sequence of myopic problems upon the arrival of each individual request using a rolling horizon approach.

Fig. 1 presents the timeline of our proposed dynamic approach. Upon the arrival of request $n$, the state of the system is shown by $X_n$ which includes fleet status such as location of each vehicle, the details of its assigned trips and a list of requests that has to be served. After the customer makes her choice from the alternatives in the optimal offered assortment, the updated status of the system is shown by $Y_n$. The notation $Y$ contains the same information as $X$ and is only used to distinguish between initial state and after request $n$ is processed. At the end of this section, we provide an illustrative example highlighting their differences.

A new request is presented by a tuple consisting of origin, destination and the preferred pickup time ($\tau_n^p$). As can be seen in Fig. 1, as soon as the new request comes in, first, we generate a set of *personalized alternatives* ($a \in A$) using different types of services and flexible time windows based on the preferred pickup time indicated by the customer (Section 4.1). Then, a lower and an upper bound for the price of each alternative is calculated (noted by $[\underline{F_a}, \overline{F_a}]$) and the *price levels* are uniformly discretized over the calculated interval (Section 4.2). In the next step, all possible sets of *assortments* ($S$) are produced for this new request by combining the price levels and the set of *personalized alternatives*. We can then reduce the size of $S$ by the methods explained in Section 6.1. Given the current state of the system ($X_n$) and set $S$, our proposed choice-driven dial-a-ride is solved with the optimal assortment ($s^*$) as output. All the alternatives in $s^*$ respect routing constraints and the customer preferences in terms of the type of service and the pickup time. CD-DARP's mathematical formulation is presented in Section 5. We need to note that the operator can reject a new request when: (i) there is no vehicle available to serve the new customer, and (ii) the expected profit is negative. In these cases, we define an empty set as the offered assortment.

Given assortment $s^*$, the customer can either choose one of the offered alternatives or decide to leave the system. Upon receiving the acceptance, we show the choice of the new customer by $\Gamma_n$, presented by a tuple including origin, destination, service type, confirmed pickup time window and the maximum ride-time ($L_n^{Max}$). At this point, we adjust the routes of all vehicles in the network (i.e., re-routing, see Section 7.1 for details). Finally, vehicles' schedules are updated for the entire network (Section 7.2). This phase is presented on the right-hand side of Fig. 1. Note that both sets of $A$ and $S$ are request-dependent and, for the sake of simplicity, we eliminated index $n$.

**Customer interaction sequences.** The system communicates information with the new customer on two occasions. First, when the customer chooses one of the offered alternatives, she is notified about her confirmed pickup time window. Second, the moment the assigned vehicle hits the road to pick her up, she is notified about the exact time the vehicle is expected to arrive.

**Graph representation.** We consider a transportation system with a set of homogeneous vehicles ($k \in K$) with a fixed capacity of $Q$. Given that customers $\{1, \dots, n-1\}$ are the ones whose requests have already been confirmed and $n$ is the new request, nodes $i$ and $n+i$ denote the pickup and drop-off locations of customer $i$. Nodes $n$ and $2n$ represent the pickup and drop-off locations of the new customer. The problem is defined on a complete directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ where $\mathcal{N} = \mathcal{P} \cup \mathcal{D} \cup \mathcal{O}$ is the set of nodes and $\mathcal{A}$ is the set of arcs. Subsets $\mathcal{P} = \{1, \dots, n\}$ and $\mathcal{D} = \{n+1, \dots, 2n\}$ denote the pickup and drop-off locations, respectively. Set $\mathcal{O}$ indicates vehicles' shift schedule: (i) the starting time and initial position of vehicle $k$ (called $o^k$), (ii) the time the shift of vehicle $k$, $\hat{o}^k$ (i.e., set $\mathcal{O} = \{o^k, \hat{o}^k | k \in K\}$) finishes. For each arc $(i,j) \in \mathcal{A}$, a routing cost $c_{ij}$ and travel time $t_{ij}$ are defined. A non-negative service time $d_i$ is defined for all pickup and drop-off nodes.

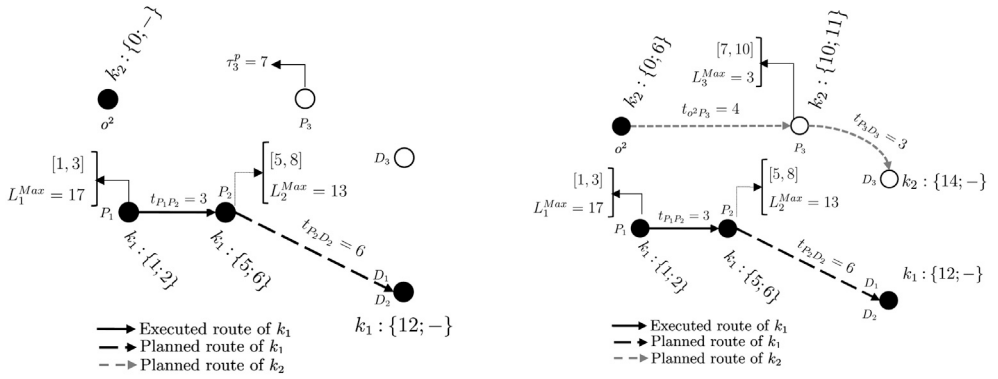
**Vehicle status.** At state $X_n$, the route of vehicle $k$ consists of a sequence of arcs with each arc being either *planned* or *executed*. An arc is defined as *executed* when a vehicle starts the journey to go to the next location. The trip assigned to the *executed* arc cannot be modified. As long as the vehicle has not departed a node (either initial or a via point), the arc is considered to be *planned*. The *planned* arcs are still subject to change, to accommodate a new request $n$. To identify whether an arc is *planned* or *executed*, we define $B_i^k$ and $\bar{B}_i^k$ as the arrival and departure time of vehicle $k$ at node $i \in \mathcal{N}$. Given $d_i$, the relationship between $B_i^k$ and $\bar{B}_i^k$ can be interpreted for two possible cases:

1. Empty vehicle: in this case, $\bar{B}_i^k = B_i^k + W_i^k + d_i$, where $W_i^k$ indicates the idle time of vehicle $k$ at node $i$.
2. Non-empty vehicle: when there are passengers on board of the vehicle, it is not allowed to wait. In other words, $\bar{B}_i^k = B_i^k + d_i$.

At state $X_n$, we consider an arc on a given route of vehicle $k$, to be *executed* if and only if $\tau_n^{arr} \geq \bar{B}_i^k$ ($i$ is the starting point of the arc). Note that the maximum ride-time of the existing requests is considered in the CD-DARP model while revising the routes (i.e., re-routing). Relocating and re-balancing idle vehicles fall outside of scope of this paper. We use Example 1 to clarify the network's setup and visualize its dynamic nature.

**Example 1.** In this example, we show how upon arrival of a new request, the current state of the system $X_n$ changes to $Y_n$ using the output of the CD-DARP. We consider a case where the operator uses two vehicles ($k_1$ and $k_2$) to serve two confirmed requests. The first request is characterized by ($P_1$, $D_1$, shared-taxi, $[1,3]$, $L_1^{Max} = 17$) in which $P_1$ and $D_1$ show the pick-up and drop-off locations. Time interval $[1,3]$ is the pick-up time window (in minutes) and, $L_1^{Max} = 17$, (in minutes) defines the maximum ride-time. Similarly, the second request is identified by ($P_2$, $D_2$, shared-taxi, $[5,8]$, $L_2^{Max} = 13$). At time $\tau_3^{arr} = 3$, the third request arrives. For this request, the pickup and drop-off locations are denoted by $P_3$ and $D_3$ and the preferred pickup time is indicated by $\tau_3^p = 7$.

Fig. 2(a) shows the state of the system upon arrival of the third request, $X_3$. For the sake of simplification, we eliminate nodes $o^1, \hat{o}^1$ and, $\hat{o}^2$. We assume that the service time for pickup and drop-off nodes is equal to one minute. In this figure, we show the schedule of each vehicle by $\{B_i^k; \bar{B}_i^k\}$. For example, at node $P_1$, schedule of vehicle $k_1$ is indicated by $\{1;2\}$. This means that vehicle




(a) State of the system upon arrival ($X_3$) (b) State of the system after confirmation ($Y_3$)

**Fig. 2.** State of the system for request 3 arrived at $\tau_3^{arr} = 3$. Black and white nodes show the pickup and drop-off locations of registered and new requests, respectively.

$k_1$ arrives at $P_1$ at time one and leaves this node at time two. Similarly, at node $o^2$, $k_2 : \{0, -\}$ shows that vehicle $k_2$ is idle. At the time of arrival of this new request, $\tau_3^{arr} = 3$, vehicle $k_1$ has already left node $P_1$ and is headed towards $P_2$. As a result, the arc connecting these two nodes is considered *executed* and cannot change. However, the arc connecting $P_2$ and $D_1$ is *planned* and could still be modified to accommodate the third request. We assume that the customer chooses a service with the following characteristics ($P_3$, $D_3$, taxi, $[7, 10]$, $L_3^{Max} = 3$) among the alternatives inside the assortment on offer. Consequently, the routes of both vehicles are updated. The state of the system is now identified by $Y_n$ as shown in Fig. 2(b).

## 4. CD-DARP elements: Assortment and pricing

One of the main characteristics of our proposed CD-DARP model lies in the definition of the objective function. In this problem, we aim at maximizing the network's expected profit which is calculated by subtracting the operational costs from the expected revenue. The latter is calculated for each alternative where the probability of one of them being selected is presented by a choice model. Operational costs is calculated according to the routing cost associated with each one the offered alternatives within the optimal assortment served by one of the available vehicles in the system.

When a new request arrives, as shown in Fig. 1, we first need to generate *personalized alternatives* (Section 4.1), then define *price levels* and finally create all possible sets of *assortments* (Section 4.2). These sets are the inputs of the CD-DARP model introduced in Section 5.

### 4.1. Personalized alternatives

We define an *alternative* ($a \in A$) as a unique combination of service type (taxi, shared taxi) and pick-up time. As mentioned in Fig. 1, alternative set $A$ is defined upon arrival of each new request. The dynamic aspect of our framework is embedded in the use of flexible pick-up time windows and service types to generate these alternatives. Given $\tau_n^p$, we generate a set of possible pick-up time windows which defines different potential delay times the customer could experience. We use $\Omega$ to be a set of potential delays. We assume that $\Omega$ is independent from the service type. For the sake of presentation, we consider three possible *delayed pickup times* (indicated by, $\Delta T$, $2\Delta T$ and $3\Delta T$) to create these flexible time windows (i.e. $\Omega = \{\Delta T, 2\Delta T, 3\Delta T\}$). For a given service type (taxi or shared taxi), they are represented by $[\tau_n^p,\ \tau_n^p + \Delta T]$, $[\tau_n^p,\ \tau_n^p + 2\Delta T]$ and, $[\tau_n^p,\ \tau_n^p + 3\Delta T]$.

**Choice model.** We assume that the probability of choosing an alternative follows a multinomial logit (MNL) function. For alternative $a \in A$ offered at price level $l \in F_a$, ($F_a$ indicates the set of *price levels* associated with alternative $a$), a utility function, $u_{al} = v_{al} + \varepsilon_a$ is defined where $v_{al}$ represents the systematic part and $\varepsilon_a$ is the stochastic part following Type I Extreme Value distribution. The systematic part of utility is presented as follows:

$$v_{al} = ASC_a - \beta^f f_{al} - \beta^t t_a - \beta^{\omega} \omega_a \qquad \forall a \in A,\ l \in F_a \tag{1}$$

In Eq. (1), $\beta^f$ shows the coefficient of the *price*. Value of $f_{al}$ shows the price level $l$ for alternative $a$ (explained in Section 4.2). $\beta^t$ presents the coefficient of *in-vehicle time* associated with $t_a$ that indicates the maximum travel time of alternative $a$ for both taxi and shared taxi. For the sake of presentation, when $a$ is a taxi service, $t_a$ is denoted by $I^{taxi}$ to indicate the shortest travel time from the origin to the destination. Using $I^{taxi}$, we pre-define a value to present the maximum ride time for a shared taxi service. In our implementation, we calculate this maximum ride time by $t_a = \min\{1.5\ I^{taxi}, I^{taxi} + 15\}$.

$\beta^{\omega}$ presents the coefficient of the *delayed pickup times*. Finally, $\omega_a$ presents the delay whose values are selected from the set $\Omega = \{\Delta T, 2\Delta T, 3\Delta T\}$. We use the maximum ride time and delayed pickup times to estimate the utility associated with each alternative. We acknowledge that utility value of each alternative is underestimated using this approach. The realized utility of the customer may exceed this estimated value.

The customer is informed about the exact waiting time, once the vehicle hits the road to the pickup location. Subsequently, the realized value of the ride time can be calculated as soon as the customer is dropped off at the destination. Both maximum ride time and delayed pickup times respect the guaranteed values embedded in the alternatives. We must clarify that the users pay according to the estimated utility when the request is made. This approach may have long-term repercussions on user choices due to the fact that it is independent from the realized value of waiting time and the maximum ride time.

The value of $ASC_a$ shows the alternative specific constant. In this problem, customers are not captive in the system, meaning that they also have the option not to choose any of alternatives being offered. In this case, we assume that the customer uses the service offered by the competitor, whose utility is identified by $u_0$. More details are discussed in Section 8.1.

### 4.2. The price of alternatives

In this problem, we assume that the operator is committed to providing attractive mobility services to the users. As a result, the price of alternatives is designed to maintain service attractiveness. In this paper, the price is set independently of resource availability and surge demand. We use the above-mentioned choice model to define the price levels of the alternatives. The choice model considers the service offered by a competitor via an opt-out option. The prices associated with each alternative in the offered assortments is determined regardless of the final choice of the customer. The customer may decide to choose one of the provided alternatives at a given price or leave the system.

We formalize our pricing strategy by defining a price range for each alternative $a$, noted by $[\underline{F_a}, \overline{F_a}]$. The lower limit $\underline{F_a}$ shows the minimum price that is still profitable for the operator. The lower price level is calculated for each given origin–destination based on the combination of a fixed and an expected variable cost of routing proportionately to the shortest path between the pair of origin–destination. A sensible upper limit is set by specifying a 'cut-off' purchasing probability. We define $\zeta_a$ to present this probability. Let $\hat{\upsilon_a}$ be a value showing the combinations of all attributes except for the price in the utility function, Eq. (1). This value is presented as follows: $\hat{\upsilon_a} = ASC_a - \beta^t t_a - \beta^{\omega}\omega_a$. In this case, the utility function can be rewritten by $\hat{\upsilon_a} - \beta^f \overline{F_a}$ where $\overline{F_a}$ is a variable representing the maximum price for alternative $a$. For each alternative, we calculate the maximum price level by solving,

$$\mathrm{Z}: \quad \max \quad \overline{F_a}: \quad \frac{e^{\hat{\upsilon_a} - \beta^f \overline{F_a}}}{e^{\hat{\upsilon_a} - \beta^f \overline{F_a}} + e^{\upsilon_0}} \geq \zeta_a$$

$\upsilon_0$ shows the opt-out option which is defined for each customer separately. Model Z can be solved analytically. Once we have the price interval (i.e. $[\underline{F_a}, \overline{F_a}]$), we define a set of price levels ($f_{al} \in F_a$) by discretizing this interval. As can be seen in Fig. 1, the last component to be calculated before solving the CD-DARP is the set of assortments. Below, we explain how these sets are generated using *personalized alternatives* and their associated *price levels*.

### 4.3. Assortment

Given the set of generated alternatives $A$ and their associated price levels $F_a$, we generate the set of all possible assortments $s \in S$. We define a binary parameter $b^s_{al}$ to be equal to 1 if alternative $a$ is offered at price level $l$ in assortment $s$. Each alternative offered in $s$ cannot take more than one price level (i.e. $\sum_{l \in F_a} b^s_{al} \leq 1$). When alternative $a$ is not offered in assortment $s$, then $\sum_{l \in F_a} b^s_{al} = 0$. The probability that a customer chooses alternative $a$ in assortment $s$ is

$$\mathbb{P}(a; s) = \frac{\sum_{l \in F_a} b^s_{al} e^{\upsilon_{al}}}{\sum_{a' \in A} \sum_{l \in F_a} b^s_{a'l} e^{\upsilon_{a'l}} + e^{\upsilon_0}} \qquad s \in S, a \in A \tag{2}$$

Below, we discuss an example where we show how the alternatives, price levels and assortments are generated when a new request arrives in the system. All this information is used as input for the CD-DARP model.

**Example 2.** A new customer arrives in the system at time $\tau^{arr}_2 = 4$. For the sake of simplicity, we assume only a taxi service is offered by the operator. The preferred pick-up time of this new request is shown by $\tau^{p}_2 = 7$. Three *delayed pick-up times* are generated accordingly. The characteristics of each alternative are shown in Table 1. For each alternative, columns 'Service', '$\omega_a$' and '$t_a$' report the service type, waiting time and travel time (shortest path), respectively. Given an estimated value for the opt-out option, the price levels between the upper and lower bound, i.e., $\underline{F_a}$, $\overline{F_a}$ as well as the utility values related to each alternative are calculated. In Table 2 we report a subset of all possible assortments given limited price levels. For each assortment, the set of available alternatives and their associated price values are reported. In the next section, we introduce the mathematical formulation of the CD-DARP.

**Table 1**
Set of taxi alternatives for the second customer.

| $a \in A$ | Service type | $\omega_a$ | $t_a$ | Delayed pickup time | $\underline{F_a}$ | $\overline{F_a}$ |
|---|---|---|---|---|---|---|
| I | Taxi | 3 | 6 | [7, 10] | 6 | 10 |
| II | Taxi | 6 | 6 | [7, 13] | 6 | 9 |
| III | Taxi | 9 | 6 | [7, 16] | 6 | 8 |

**Table 2**
Subset of generated assortments for Example 2.

| Assortments | Alternatives | | | | | |
|---|---|---|---|---|---|---|
| | Available | | | Price | | |
| | I | II | III | I | II | III |
| $s_1$ | No | No | No | – | – | – |
| $s_2$ | Yes | No | No | 6 | – | – |
| $s_3$ | Yes | No | No | 7 | – | – |
| $s_4$ | No | Yes | No | – | 8 | – |
| $s_5$ | Yes | Yes | No | 8 | 7 | – |
| $s_6$ | No | Yes | Yes | – | 8 | 7 |
| $s_7$ | Yes | Yes | Yes | 9 | 8 | 7 |

## 5. CD-DARP mathematical model

We first present the mathematical model in its nonlinear form. The linearization of this formulation is presented in Section 5.1. As mentioned in Section 3, we model the problem on a complete directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ where nodes $n$ and $2n$ represent the pick-up and drop-off locations of the new customer. For assortment $s \in S$, we define $\mathbb{E}_s$ as its expected profit and the binary variable $y_s = 1$ if it is offered. The CD-DARP problem is presented as follows:

$$\max \quad \sum_{s \in S} \mathbb{E}_s y_s \tag{3}$$

st.

(4)–(37)

The objective function (3) calculates the difference between the revenue generated from offering assortment $s$ and its associated cost. Routing, time window, vehicle load and assortment constraints are presented by (4)–(37). In the remainder of this section, we explain each family of constraints separately.

**Marginal routing cost.** Expected profit in (3) has two components: generated revenue as a result of offering assortment $s$ and the associated costs the definition of which lies in the introduction of a value called *marginal routing cost*. Let $\bar{C}^k$ present the routing cost of vehicle $k \in K$ before arrival of request $n$. When request $n$ arrives, we have the possibility of modifying vehicles' routes to accommodate it. Thus, the routing cost of vehicle $k$ is $\sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}} c_{ij} x_{ij}^k$. Fractional variable $\sigma^k$ presents the marginal cost of vehicle $k$ after the new customer arrives which is given by (4).

$$\sigma^k = \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}} c_{ij} x_{ij}^k - \bar{C}^k \qquad k \in K, \tag{4}$$

$$\sigma^k \in \mathbb{R} \qquad k \in K, \tag{5}$$

Each one of the alternatives in the assortment being offered is assigned to a given vehicle. The cost of this alternative is calculated based on the marginal routing cost of the assigned vehicle (i.e., Constraint (4)). We define $\delta_a$ as a fractional variable denoting the cost of alternative $a$ in the offered assortment. Binary variable $z_a^k$ is equal to one when vehicle $k$ is used to serve alternative $a$. Constraints (6) show if vehicle $k$ is assigned to alternative $a$ (conditioned to the fact that alternative $a$ is in the offered assortment), the cost of this alternative is equal to the marginal routing cost of the assigned vehicle. Note that, it is possible for vehicle $k$ to be potentially assigned to multiple alternatives.

$$\delta_a = \sum_{k \in K} \sigma^k z_a^k \qquad a \in A, \tag{6}$$

$$\delta_a \in \mathbb{R} \qquad a \in A, \tag{7}$$

On the other hand, when vehicle $k$ is not assigned to serve one of the alternatives offered in the assortment, its marginal routing cost still has an impact on the overall expected profit of the assortment and must be addressed. In this case, its marginal routing cost is indicated by $\Delta^k$. For vehicle $k$, when $\sum_{a \in A} z_a^k \geq 1$ (or equivalently $\sum_{j \in \mathcal{N}} x_{nj}^k = 1$), then $\Delta^k = 0$; otherwise, $\Delta^k = \sigma^k$ ($x_{nj}^k$ indicates that vehicle $k$ is used to pickup the new passenger from its origin location). In other words, if $\Delta^k = 0$ then either the route of vehicle $k$ has remained unchanged or this vehicle is used to serve one of the alternatives in the assortment. In the later case, the marginal

routing cost of the alternative $a$ is noted by $\delta_a$. The above description is summarized by Constraints (8).

$$\Delta^k = \sigma^k(1 - \sum_{j\in\mathcal{N}} x^k_{nj}) \qquad k \in K, \tag{8}$$

$$\Delta^k \in \mathbb{R} \qquad k \in K, \tag{9}$$

**Expected profit.** Using the *marginal routing cost* discussed above, here, we present the expanded version of the objective function (3). Assume that alternative $a$ is offered in the assortment $s$ and $f_{al} \in F_a$ is the price level $l$ associated with this alternative. The profit of this alternative is then defined by $r_a = f_{al} - \delta_a$. Given Eq. (2), $\mathbb{P}(a;s)$ shows the probability of selecting alternative $a$ in assortment $s$. As a result, the expected profit of offering assortment $s$ (shown by $\mathbb{E}_s$) is estimated by:

$$\mathbb{E}_s = \sum_{a\in A} r_a \mathbb{P}(a;s) - \sum_{a\in A} \mathbb{P}(a;s) \sum_{k\in K} \Delta^k \qquad \forall s \in S, \tag{10}$$

In Eq. (10), the first term presents the expected profit of offering assortment $s$ summed over all its alternatives, while the second term shows the expected cost related to the re-routing of vehicles that are not used to offer any of these alternatives. The term $\sum_a P(a;s)$ shows the probability that a user chooses one of the alternatives in the assortment. If vehicle $k$ is used to serve alternative $a$, its associated cost is directly incorporated into the assortment by Constraints (6) $\delta_a = \sum_{k\in K} \sigma^k z^k_a$. If this is not the case and vehicle $k$ is not assigned to serve a request, then still the expected cost of re-routing is calculated based on the probability of choosing any alternative in the offered assortment. This estimated expected cost of re-routing indeed depends on the customer's choice. This value is calculated once prior the decision being made to determine the optimal assortment $s^*$ and once after the choice has been made to modify the vehicles' routes according to the customer's decision.

**Routing constraints.** We define the binary variable $x^k_{ij} = 1$, if vehicle $k$ travels from node $i$ to $j$. Constraints (11)–(16) determine the routing decisions. Constraints (11) make sure that all confirmed requests (i.e. $\mathcal{P}\setminus\{n\}$) are visited by exactly one vehicle. We have to remind that the new request (node $n$) can be potentially visited by several vehicles. Flow conservation constraints are presented in (12). Constraints (13) ensure that the associated pickup and drop-off nodes are visited by the same vehicle. Constraints (14)–(15) make sure that the vehicles' routes exist (i.e. vehicles are active in their scheduled shift).

$$\sum_{k\in K}\sum_{j\in\mathcal{N}} x^k_{ij} = 1 \qquad i \in \mathcal{P}\setminus\{n\}, \tag{11}$$

$$\sum_{j\in\mathcal{N}} x^k_{ji} - \sum_{j\in\mathcal{N}} x^k_{ij} = 0 \qquad i \in \mathcal{P}\cup\mathcal{D}, k \in K, \tag{12}$$

$$\sum_{j\in\mathcal{N}} x^k_{ij} - \sum_{j\in\mathcal{N}} x^k_{n+i,j} = 0 \qquad i \in \mathcal{P}, k \in K, \tag{13}$$

$$\sum_{j\in\mathcal{N}} x^k_{o^k,j} = 1 \qquad k \in K, \tag{14}$$

$$\sum_{i\in\mathcal{N}} x^k_{i,\hat{o}^k} = 1 \qquad k \in K, \tag{15}$$

$$x^k_{ij} \in \{0,1\} \qquad i,j \in \mathcal{N}, k \in K, \tag{16}$$

**Time window constraints.** As mentioned in Section 3, the arrival and departure time of vehicle $k$ at node $i$, are denoted by $B^k_i$ and $\bar{B}^k_i$, respectively. A time window $[e_i, l_i]$ and a non-negative service duration $d_i$ ($d_i = 0\ \forall i \in \mathcal{O}$) are defined for node $i \in \mathcal{N}\setminus\{n, 2n\}$. Time windows associated with the nodes in the set $\mathcal{O}$ represent the availability of vehicles. At node $i \in \mathcal{N}$, Constraints (17) and (18) track the travel time, ($M_{ij}$ is a large constant value).

$$B^k_j \geq \bar{B}^k_i + t_{ij} - M_{ij}(1 - x^k_{ij}) \qquad i,j \in \mathcal{N}, k \in K, \tag{17}$$

$$\bar{B}^k_i \geq B^k_i + d_i \qquad i \in \mathcal{N}, k \in K, \tag{18}$$

$$B^k_i, \bar{B}^k_i \in \mathbb{R}^+ \qquad i \in \mathcal{N}, k \in K, \tag{19}$$

For the confirmed requests, Constraints (20) present the time window restrictions. However, as previously mentioned for the new request, several flexible time windows are generated (related to the pick-up node $n$ and drop-off node $2n$). $[e_{ia}, l_{ia}]$ shows the time window of alternative $a$. When vehicle $k$ is assigned to alternative $a$, (i.e., $z^k_a = 1$), Constraints (21) ensure that the vehicle respects these time window restrictions. Later, we explain how $M^l$ and $M^u$ (both large constant values) are defined.

$$e_i \leq B^k_i \leq l_i \qquad i \in \mathcal{N}\setminus\{n,2n\}, k \in K, \tag{20}$$

$$e_{ia} - M^l(1 - z^k_a) \leq B^k_i \leq l_{ia} + M^u(1 - z^k_a) \qquad i \in \{n,2n\}, k \in K, a \in A, \tag{21}$$

$$z^k_a \in \{0,1\} \qquad a \in A, k \in K, \tag{22}$$

We introduce variable $L^k_i$ as the ride time of request $i$ on vehicle $k$. Constraints (23) define the ride time of each user and makes sure that the drop-off node is visited after the pick-up node. The limit on maximum ride time is ensured by Constraints (24). Note that similar maximum ride time is defined for all shared-taxi alternatives.

$$L^k_i = B^k_{n+i} - (B^k_i + d_i) \qquad i \in \mathcal{P}\cup\{n\}, k \in K, \tag{23}$$

$$L_i^k \le L_i^{Max} \qquad i \in \mathcal{P}, k \in K, \tag{24}$$

$$L_i^k \in \mathbb{R}^+ \qquad i \in \mathcal{P}, k \in K, \tag{25}$$

**Vehicle load constraints.** We define parameters $q_i = -q_{n+i}$ , $i \in \mathcal{P} \setminus \{n\}$ to indicate the load of node $i$. It is worth mentioning that $q_i = 0$ , $\forall i \in \mathcal{O}$ (i.e., initial position of the vehicle). This value also presents the service type. At node $i \in \mathcal{P} \setminus \{n\}$, if taxi service is offered then we have $q_i = Q$, otherwise $q_i = 1$ ($Q$ is the capacity of each vehicle). We model the load associated with the new request by a variable named $q'^k$ whose value can either take 1 or $Q$ depending on the type of service. Variable $Q_i^k$ is defined as the load of vehicle $k$ after visiting node $i$. (26)–(28) present the vehicle load constraints for the confirmed requests. Constraints (29) and (30) determine the vehicle load after being assigned to serve an alternative offered to the new request ($A^T \subset A$ presents set of taxi alternatives).

$$Q_j^k \ge Q_i^k + q_j - M(1 - x_{ij}^k) \qquad i \in \mathcal{N}, j \in \mathcal{N} \setminus \{n, 2n\}, k \in K, \tag{26}$$

$$Q_n^k \ge Q_i^k + q'^k - M(1 - x_{in}^k) \qquad i \in \mathcal{N}, k \in K, \tag{27}$$

$$Q_{2n}^k \ge Q_i^k - q'^k - M(1 - x_{i,2n}^k) \qquad i \in \mathcal{N}, k \in K, \tag{28}$$

$$z_a^k \le q'^k \qquad a \in A, k \in K, \tag{29}$$

$$Q z_a^k \le q'^k \qquad a \in A^T, k \in K, \tag{30}$$

$$Q_i^k \in [0, Q] \qquad i \in \mathcal{N}, k \in K, \tag{31}$$

$$q'^k \in [0, Q] \qquad k \in K. \tag{32}$$

**Assortment constraints.** Constraints (33)–(37) link the assortment and routing decisions. Parameter $\gamma_{as}$ takes value 1, if alternative $a$ is offered in $s$. Constraints (33) make sure that only one assortment can be offered in response to the new request. If assortment $s$ is offered, then each alternative in the assortment must be served by exactly one vehicle, Constraints (34). Constraints (35) and (36) link the vehicle assignment decisions with routing. If vehicle $k$ visits the pickup location of the new customer (node $n$), then it has to be assigned to at least one of the alternatives in the assortment as stated by Constraints (35). Similarly, if vehicle $k$ is assigned to an alternative, it has to visit the node associated to the new customer, Constraints (36).

$$\sum_{s \in S} y_s = 1 \tag{33}$$

$$\sum_{k \in K} z_a^k = \sum_{s \in S} \gamma_{as} y_s \qquad a \in A, \tag{34}$$

$$\sum_{j \in \mathcal{N}} x_{nj}^k \le \sum_{a \in A} z_a^k \qquad k \in K, \tag{35}$$

$$z_a^k \le \sum_{j \in \mathcal{N}} x_{nj}^k \qquad k \in K, a \in A, \tag{36}$$

$$y_s \in \{0, 1\} \qquad s \in S, \tag{37}$$

In the following subsection, we discuss the linearization of the above-mentioned nonlinear constraints and the objective function.

### 5.1. Linearization

The CD-DARP model presented by (3)–(37) is nonlinear due to Constraints (6), (8) and, the objective function (3). By introducing large constants $M$ and $\hat{M}^k$, Constraints (6) can be linearized as follows:

$$\sigma^k - M(1 - z_a^k) \le \delta_a \qquad k \in K, a \in A, \tag{38}$$

$$-\hat{M}^k \sum_{k \in K} z_a^k \le \delta_a \qquad a \in A, \tag{39}$$

$$\delta_a \le M \sum_{k \in K} z_a^k \qquad a \in A, \tag{40}$$

In a similar fashion, Constraints (8) are replaced by its linearized form presented below.

$$\sigma^k - M \sum_{j \in \mathcal{N}} x_{nj}^k \le \Delta^k \qquad k \in K, \tag{41}$$

$$-\hat{M}^k(1 - \sum_{j \in \mathcal{N}} x_{nj}^k) \le \Delta^k \qquad k \in K, \tag{42}$$

$$\Delta^k \le M(1 - \sum_{j \in \mathcal{N}} x_{nj}^k) \qquad k \in K, \tag{43}$$

Finally, by introducing variable $E \ge 0$, the objective (3) is replaced by (44)–(46).

$$\max \quad E \tag{44}$$

$$E \le \mathbb{E}_s + M(1 - y_s) \qquad s \in S, \tag{45}$$

$$E \in \mathbb{R}^+ \tag{46}$$

In the following section, we explain several of the properties of the CD-DARP that make our resolution approach more computationally efficient.

## 6. Assortment and variable reduction

CD-DARP combines a dial-a-ride problem and assortment optimization (see, Cordeau (2006) and Kök et al. (2008)). As explained in Fig. 1, before solving the CD-DARP, using the combination of *personalized alternatives* and the *price levels*, we generate all possible sets of *assortments*, see Section 4.3. This can be computationally exhaustive. In Section 6.1, we introduce several properties of the model that makes it possible to decrease the size of the set of assortments. Next, we introduce variable reduction techniques and valid inequalities to reduce the search space for computational efficiency.

### 6.1. Selecting dominant assortments

The objective function presented in (3), requires to evaluate $\prod_{a\in A}(|F_a|+1)$ assortments which is computationally cumbersome. $F_a$ presents the set of price levels associated with alternative $a$. This section proposes two policies to limit the collection of assortments offered to each arriving customer. Numerical results associated with the proposed policies compared with the complete enumeration are presented in Section 8.2.

For the ease of presentation, we call the term $\sum_{a\in A}(\delta_a + \sum_{k\in K} \Delta^k)$ *assortment cost* obtained by reorganizing (10), as follows:

$$\mathbb{E}_s = \sum_{a\in A} \mathbb{P}(a;s) f_{al} - \sum_{a\in A} \mathbb{P}(a;s)(\delta_a + \sum_{k\in K} \Delta^k). \tag{47}$$

We call two assortments *comparable* if both contain the same alternatives but at different price levels. Based on (47), all comparable assortments have identical *assortment costs* when their corresponding expected profit is maximized. This is the basis for our proposed policies.

**Policy I.** In the first policy, we assume that the impact of the *assortment cost* is negligible and exclude it from (47). Therefore, for each set of *comparable* assortment, we aim at finding the one that maximizes the revenue.

Let $\bar{A}'$ be the set of available alternatives inside the *comparable* assortments. In fact, these alternatives have the same specifications except for the price. We define the binary variable $h_{al} = 1$ if alternative $a \in \bar{A}'$ is offered at price level $l \in F_a$. The dominant assortment is determined by solving the fractional binary optimization model presented in (48).

$$\begin{aligned} \mathrm{J}: \max \quad & \frac{\sum_{a\in\bar{A}'}\sum_{l\in F_a} f_{al} e^{v_{al}} h_{al}}{\sum_{a\in\bar{A}'}\sum_{l\in F_a} e^{v_{al}} h_{al} + e^{v_0}} && \\ & \sum_{l\in F_a} h_{al} = 1 && a \in \bar{A}', \\ & h_{al} \in \{0,1\} && a \in \bar{A}', l \in F_a. \end{aligned} \tag{48}$$

Model J shares two similar properties with the optimal line selection problem discussed by Chen and Hausman (2000): (i) the objective function is both strictly quasi-convex and strictly quasi-concave, and (ii) the coefficient matrix is completely unimodular. As such, for this model, any local maximum solution is the global maximum and the optimal solution of its relaxation (by dropping the integrality constraints) is integral.

Because of these properties, the optimal solution of model J is reduced to solving its relaxation counterpart, which is a fractional optimization model. By using Charnes–Cooper transformation, we can easily transform the relaxation counterpart into a linear programming (LP) model, Charnes and Cooper (1962). The solution of the LP model finds the assortment that maximizes the expected revenue among all comparable assortments.

**Policy II.** Based on the second policy, the set of comparable assortments contain more than one assortment, i.e., indicating alternatives at different price levels.

Using the definition of the probability function presented in Eq. (2), and $v_0 = 0$, the profit maximizer assortment in the set of comparable assortments can be found as follows,

$$max\{\lambda \in \mathbb{R} : \sum_{l\in F_a | a\in\bar{A}'} h_{al} = 1, \text{and } \sum_{l\in F_a}\sum_{a\in\bar{A}'} h_{al} e^{v_{al}}(f_{al} - \delta_a - \sum_{k\in K}\Delta^k - \lambda) \ge \lambda\}$$

From geometrical perspective, inspired from the work of Rusmevichientong et al. (2010), we define a linear function named, $\eta_{al} : \mathbb{R} \to \mathbb{R}$ such that $\eta^0_{al}(\lambda) = 0$ and $\eta_{al}(\lambda) = e^{v_{al}}(f_{al} - \delta_a - \sum_{k\in K}\Delta^k - \lambda)$.

Consider the revenue maximizing problem by setting the *assortment cost* to zero. To find the optimal assortment, following from Proposition 1 in Talluri and van Ryzin (2004), we need to determine the non-dominated set of assortment for every $\lambda$. As discussed in Section 2.1 of Rusmevichientong et al. (2010), instead of finding the non-dominated set for every value of $\lambda$, we can limit our evaluations to the intersection points among all lines defined by function $\eta_{al}(\lambda)$.

Using the definition of an *efficient set* presented in Talluri and van Ryzin (2004), the efficient set can be obtained by solving model (49).

$$\max \quad \frac{\sum_{a\in\bar{A}'}\sum_{l\in F_a}(f_{al}-\lambda)e^{v_{al}}h_{al}}{\sum_{a\in\bar{A}'}\sum_{l\in F_a}e^{v_{al}}h_{al}+1} \tag{49}$$

$$\sum_{l\in F_a} h_{al} = 1 \qquad a\in\bar{A}',$$

$$h_{al}\in\{0,1\} \qquad a\in\bar{A}', l\in F_a.$$

In Policy II, the collection of assortments includes all non-dominated assortments obtained by solving model (49) at each intersection point. In model (49), we only consider the price levels in which $f_{al}-\lambda\geq 0$. In case of infeasible solution, no assortment is considered at a given intersection point.

Finally, we have to mention that for any $\sum_{k\in K}\Delta^k\geq 0$, all the intersection points will be shifted by $\sum_{k\in K}\Delta^k$. Similar argument is valid when $\delta_a=\delta_{a'}$, for all $a,a'\in\bar{A}'$.

### 6.2. Variable reduction

In Section 6.1, we presented our approach to decrease the number of variables associated with assortments (i.e. $y_s$). Let $R(\bullet)$ be a subset of arcs in the network. By some logical analysis, the following arcs can be removed from the graph ($\sum_{i,j\in R(\bullet)}\sum_{k\in K}x_{ij}^k=0$).

- $R(1)=\{i,j|j\in\mathcal{O}\cup\mathcal{D}, i=o^k, j\neq\hat{o}^k\}$,
- $R(2)=\{i,j|i\in\mathcal{O}\cup\mathcal{P}, i\neq o^k, j=\hat{o}^k\}$,
- $R(3)=\{i,j|i\in\mathcal{P}\setminus\{n\}, j\in\mathcal{N}, j\neq n+i, i \text{ is a taxi service}\}$,
- $R(4)=\{i,j|i\in\mathcal{N}, j\in\mathcal{D}\setminus\{2n\}|i\neq n, j \text{ is a taxi service}\}$,
- $R(5)=\{i,j|i,j\in\mathcal{N}, e_i+d_i+t_{ij}>l_j\}$,
- $R(6)=\{i,j,n+i|i\in\mathcal{P}, j\in\mathcal{N}, t_{ij}+d_j+t_{j,n+i}>L_i^{Max}\}$.

Let $(i,n+i)$ and $(j,n+j)$ be the corresponding nodes to two confirmed shared-taxi services. We define a path $P(\bullet)$ as a sequence of visiting nodes. By combining constraints related to the maximum ride time and time windows, the following variables can be removed from the graph if the following paths are infeasible (see, Cordeau (2006) and Dumas et al. (1991)):

- $\sum_{k\in K}x_{i,n+j}^k=0$ if $P(1)=\{j,i,n+j,n+i\}$ is infeasible.
- $\sum_{k\in K}x_{n+i,j}^k=0$ if $P(2)=\{i,n+i,j,n+j\}$ is infeasible.
- $\sum_{k\in K}x_{i,j}^k=0$ if $P(3)=\{i,j,n+i,n+j\}$ and $P(4)=\{i,j,n+j,n+i\}$ are infeasible.
- $\sum_{k\in K}x_{n+i,n+j}^k=0$ if $P(5)=\{i,j,n+i,n+j\}$ and $P(6)=\{j,i,n+i,n+j\}$ are infeasible.

### 6.3. Setting delivery time window

In CD-DARP, time windows are defined only for the pick-up locations. Let $i\in\mathcal{P}$ be a pick-up node with the time window $[e_i,l_i]$. If node $i$ is a confirmed taxi service, then the associated delivery time window is determined by: $[e_i+t_{ij},l_i+t_{ij}]$ where $t_{ij}$ is the minimum travel time from pick-up $i$ to delivery $j$. On the other hand, if the service assigned to node $i$ is confirmed to be a shared-taxi, the delivery time window is set as $[e_i+t_{ij},l_i+L_i^{Max}]$. The time window associated with the delivery of a new request is calculated using the same rule that is applied to the shared-taxi.

### 6.4. Setting values for large constants

Constraints (39) and (42) define lower bounds for the marginal cost. This bound is imposed by setting $\hat{M}^k=\bar{C}^k$ (previously defined in Section 5.1). In Constraints (17), $M_{ij}$ is set as the $\max\{0,l_i+d_i+t_{ij}-e_i\}$. For the nodes $i\in\{n,2n\}$, we define $\underline{e_i}=min_{a\in A}\{e_{ia}\}$, $\overline{e_i}=max_{a\in A}\{e_{ia}\}$, $\underline{l_i}=min_{a\in A}\{l_{ia}\}$, $\overline{l_i}=max_{a\in A}\{l_{ia}\}$. The value of $M^u$ and $M^l$ in Constraints (21) is obtained by setting $M^u=\overline{l_i}-\underline{l_i}$ and $M^l=\overline{e_i}-\underline{e_i}$.

### 6.5. Infeasible path inequalities

As mentioned earlier, $\tau_n^{arr}$ shows the arrival time of customer $n$ and $\bar{B}_i^k$ as the departure time of assigned vehicle $k$ from node $i$. We define $\bar{\mathcal{N}}=\{i\in\mathcal{N}|\bar{B}_i^k>\tau_n^{arr}\}$ as the set of nodes which has not yet been served. Let $(i,n+i)\in\bar{\mathcal{N}}$ be a pick-up candidate and its associated delivery node in this set. As proposed by Cordeau (2006), any directed path $\mathsf{P}=\{i,j',\ldots,j'',n+i\}$ in which $j',j''\in\bar{\mathcal{N}}$ such that its duration is greater than a maximum ride time (i.e. $t_{i,j'}+d_{j'}+\cdots,t_{j'',n+i}>L_i^{Max}$) is valid for CD-DARP. In our implementation, we add limited number of infeasible paths to the model in advance. In this paper, instead of using path construction heuristic proposed by Cordeau (2006), we verify all the possible paths. To control the procedure and number of generated valid inequalities, we limited our search to the paths whose *length* is less than or equal to three, mainly because of the capacity of vehicles. Let $\mathsf{P}$ be the path that violates the maximum ride time constraint. If $n$ or $2n$ belong to $\mathsf{P}$, then we add $\sum_{i,j\in\mathsf{P}}x_{ij}^k\leq length-1\ \forall k\in K$ to the model; otherwise, the inequality $\sum_{k\in K}\sum_{i,j\in\mathsf{P}}x_{ij}^k\leq length-1$ is added.

## 7. Rolling horizon approach

As mentioned in Section 3, customers arrive in the system one at a time. Each time a new customer arrives, the operator solves the CD-DARP to offer the most convenient (for the customer based on his or her preferences) and profitable (for the network) assortment. Algorithm 1 summarizes the rolling horizon approach. When a new request arrives, given the current status of vehicles ($Y_{n-1}$) and the time of arrival of this new request ($\tau_n^{arr}$) the current state of the system is retrieved ($X_n$). Based on the characteristics of the new request, such as, pick-up and drop-off locations, as well as the preferred pickup time ($\tau_n^p$), the *personalized alternatives* ($A$) are generated. Next, the utility of opt-out option ($v_0$) and the *price levels* ($F_a$) of each alternative are defined, (steps 1 to 6). In step 7, set of all *assortments* ($S$) are introduced after which, given ($S$, $A$, $X_n$), the CD-DARP is calculated whose output is the optimal assortment ($s^*$) to be offered (steps 8 and 9). As soon as the customer makes a choice ($\Gamma_n$) among the available alternatives in $s^*$, the routes of all vehicles and their associated schedules are modified (steps 10 and 11).

In Section 7.1, we explain how the routes of vehicles are updated as soon as a choice is made by the new customer. Then, in Section 7.2, we discuss how the vehicles' schedules (dispatch times) are updated, before introducing the numerical results in Section 8.

**Algorithm 1** Rolling horizon approach

*Initialize*

$Y_0 \leftarrow$ Get the initial location of vehicles ($K$)*

1 **while** the operator is active **do**
2 Upon arrival of request $n$ **do**
3 $X_n \leftarrow$ Get the status of vehicles ($Y_{n-1}, \tau_n^{arr}$) — Section 3
4 $A \leftarrow$ Generate alternatives (pickup, drop-off location of request $n$, $\tau_n^p$) — Section 4.1
5 $v_0 \leftarrow$ Estimate the opt-out option (pickup, drop-off location of request $n$) — Section 4.1
6 $F_a \leftarrow$ Set the price of alternatives ($A, v_0$) — Section 4.2
7 $S \leftarrow$ Get the set of assortments ($A, F_a$) — Section 6.1
8 $s^* \leftarrow$ Solve the CD-DARP ($S, A, X_n$) — Sections 5 and 6
9 Offer $s^*$ to the customer
10 $\Gamma_n \leftarrow$ Receive the choice of customer ($s^*$)
11 $Y_n \leftarrow$ Modify the routes of all vehicles and update vehicles' schedules ($\Gamma_n$), — Sections 7.1 and 7.2

* input information for each step indicated inside the parenthesis

### 7.1. Updating vehicles routes: re-routing

According to Fig. 1, as soon as the new customer makes a decision ($\Gamma_n$), vehicle routes need to be updated (re-routing). As mentioned in Section 5, the optimal vehicle routes are obtained before the arrival of a new customer to calculate the marginal cost indicated by $\bar{C}^k$. When solving the CD-DARP, we allow for multiple vehicle visits associated with the new customer. Therefore, the routing decision of solving CD-DARP may violate the optimal route assumption after the choice is made. As a result, we solve the inherit DARP model to modify the vehicle routes. The CD-DARP solution (Step 8) contains information regarding the assortment and vehicle routes. After receiving the customers' decision, we extract the information about the assignment of vehicles to the requests from the CD-DARP solution obtained in step 8. We use this information in step 11 to speed up solving the DARP model.

### 7.2. Updating vehicles' schedules

Vehicle dispatch time is the moment when the *planned route* is set to be *executed*. Vehicle dispatching strategy affects the system's performance, e.g., routing cost. (see, Thomas (2007) and Mitrovic-Minic et al. (2004)). However, the main challenge is to decide at which time the vehicle needs to leave a certain location. As suggested by Mitrovic-Minic et al. (2004), it is better to distribute the available waiting time through the planned routes. This strategy leads to route cost reduction, as it provides opportunities to re-optimize them.

Due to the nonlinearity of time tracking constraints (which have been linearized in our case presented by Constraints (17)–(18)), calculating vehicles' waiting time is not straight-forward, see, Savelsbergh (1992). The main challenge is to respect the service time and the maximum ride time of the existing customers in the network. Also, it requires avoiding delays at a given location when the vehicle is loaded with passengers. For the planned route of vehicle $k$, we define $\bar{\mathcal{P}}^k$ as a set of pickup nodes. Similarly, set $\bar{\mathcal{N}}^k$ denotes the set of all visited nodes ($\bar{\mathcal{P}}^k \subseteq \bar{\mathcal{N}}^k$). As mentioned before, for node $i$ and vehicle $k$, we define $W_i^k$ to be the waiting time. The dispatch time of each vehicle can be determined by solving the following LP model (50)–(54).

$$\max \quad \sum_{i \in \bar{\mathcal{N}}^k | i \neq \hat{o}^k} \sum_{k \in K} W_i^k \tag{50}$$

$$W_i^k + d_i = \bar{B}_i^k - B_i^k \qquad k \in K, i \in \bar{\mathcal{N}}^k, \tag{51}$$

**Table 3**
Characteristics of the test instances.

| Inst. name | Service zone | Trip dens. | Booking behavior | Trip length (km) | | | |
|---|---|---|---|---|---|---|---|
| | | | | $\mu$ | $\sigma$ | Min | Max |
| 10-US | 10 × 10 | **U** | $U(5,10)$ | 5.34 | 2.31 | 1.15 | 11.27 |
| 10-UM | | | $U(5,15)$ | | | | |
| 10-UL | | | $U(10,15)$ | | | | |
| 10-CS | | **C** | $U(5,10)$ | 3.68 | 2.10 | 1.10 | 11.32 |
| 10-CM | | | $U(5,15)$ | | | | |
| 10-CL | | | $U(10,15)$ | | | | |
| 15-US | 15 × 15 | **U** | $U(5,10)$ | 7.69 | 3.60 | 1.57 | 17.03 |
| 15-UM | | | $U(5,15)$ | | | | |
| 15-UL | | | $U(10,15)$ | | | | |
| 15-CS | | **C** | $U(5,10)$ | 5.08 | 3.01 | 1.18 | 17.72 |
| 15-CM | | | $U(5,15)$ | | | | |
| 15-CL | | | $U(10,15)$ | | | | |

$$B_{i+n}^k - B_i^k - d_i \leq L_i^{Max} \qquad k \in K, i \in \bar{P}^k, \tag{52}$$

$$e_i \leq B_i^k \leq l_i \qquad k \in K, i \in \bar{\mathcal{N}}^k, \tag{53}$$

$$W_i^k = 0 \qquad k \in K, i \in \bar{\mathcal{N}}^k | Q_i^k \neq 0. \tag{54}$$

The objective function (50) maximizes the total waiting time. The waiting time at node $i$ is defined by Constraints (51). Constraints (52) and (53) make sure that the maximum ride time of each request and their time windows are respected. Constraints (54) prohibit additional waiting time with passengers on board.

## 8. Numerical results

We conduct our experiments using C++ and CPLEX 12.8. All experiments are carried out on a computer with a 2.4 GHz CPU and 8 GB of RAM. In Section 8.1, we outline data generation scheme and customer simulation setting. In Section 8.2, we investigate the computational performance of our model. Next, in Section 8.3, we evaluate the impact of introducing flexible pick-up time windows on the system's performance. By allowing for this flexibility, we can significantly improve the performance of the system (i.e. the number of passengers served while reducing total routing cost). In Section 8.4, we show the trip-based pricing improves the profit performance as well as customer acceptance compared to the flat rate pricing. Finally, in Section 8.5, we test the proposed algorithm on New-York green taxi data and compare it with the current practice. The results demonstrate the benefits of dynamic CD-DARP in practice.

### 8.1. Instance description

We assume that all trips are performed inside a predefined service zone. Two service zones are identified: an area of 10 × 10 km$^2$ and 15 × 15 km$^2$, respectively. Furthermore, two scenarios are considered to show how trips are distributed within the service zone. In the first scenario, (noted by **U**), origin and destination nodes are uniformly distributed. The second scenario (**C**) presents a situation in which 75% of the trips either originate or end in the city center. We present the city center as a circle with a radius of **r** km originated at the center of the service zone (**r** = 2.5 and **r** = 4.5 km for 10 × 10 and 15 × 15 km$^2$ instances, respectively). We then uniformly generate nodes 75% of which are positioned within the radius of **r** km. We randomly label the nodes that are generated as origin and destination, to create trips. We discard trips where the Euclidean distance is less than one kilometer. For all instances, we consider a set of homogeneous vehicles with the capacity ($Q$) of three.

**Data generation.** The experiments are set up for two peak hour intervals, one in the morning and one in the afternoon, in minutes $[75, 225]$ and $[345, 450]$. We assume that, within these time periods, customers arrive randomly according to a non-homogeneous Poisson process with an inter-arrival time of four and two minutes for off-peak and peak hours, respectively. Note that the assumption of random arrivals according to this Poisson process is only used for generating instances and our proposed CD-DARP model assumes no prior information regarding the customer arrivals. Moreover, for the sake of fair comparison across the instances, we only consider the first 150 customers. Instances are generated according to three main attributes: (1) trip length, either short or long, (2) trip distribution (i.e., the trip requests can be either uniformly distributed in the area or concentrated towards the center of the region) and, finally, (3) customer booking behavior in terms of their tolerance against possible delayed pick-up time. The characteristics of the instances are shown in Table 3.

As suggested by Bösch et al. (2018), we set the transportation cost at $ 0.41 per km. For each instance, we vary the number of vehicles (two, four and six) to examine the trade-off between resource availability and demand volume. The initial location of the vehicles is randomly determined within the service area. All vehicles are available during the service period. For each request, we define the preferred pickup time ($\tau_i^p$) as $\tau_i^p = \tau_i^{arr} + \tau$ in which $\tau_i^{arr}$ is the arrival time of the customer $i$ and $\tau$ is a random variable taken from a uniform distribution ($\tau_i^p > \tau_i^{arr}$). Here, we define three cases to evaluate people's booking behavior: (i) impatient

**Table 4**
Choice model parameters, Krueger et al. (2016).

| Attributes | Taxi | Shared-Taxi | Public transport |
|---|---|---|---|
| ASC | −0.28 | −0.77 | 0 |
| In-vehicle time ($\beta^t$) | −0.792 | −0.85 | −0.88 |
| Delay ($\beta^o$) | −0.06 | −0.10 | −0.0356 |
| Price ($\beta^f$) | −1.2 | −1.04 | −0.89 |

**Table 5**
Values used in the dynamic CD-DARP.

| Description | Value | Description | Value |
|---|---|---|---|
| Delay ($\Delta t$) | 3 min | Service time ($d_i$) | 0.5 min |
| Vehicle speed | 50 km/h | Public transport speed | 20 km/h |
| Min purchase probability ($\zeta$) | 0.05 | Price discretization level | 0.1 |
| Vehicle capacity ($Q$) | 3 | Transportation cost per km | 0.41 $ |

customers ($\tau \sim U(5, 10)$), shown by **S** in Table 3, (ii) patient customers ($\tau \sim U(10, 15)$), shown by **L** in Table 3, and (iii) a mixture of both groups ($\tau \sim U(5, 15)$), shown by **M** in Table 3. The summary of the attributes for each instance is shown in Table 3, including 12 instances. For example, **10-US** shows the area of $10 \times 10$ km$^2$ in which the demand is uniformly distributed (**U**) and customers are impatient (**S**). In this table, four columns under the trip length show the average ($\mu$), standard deviation ($\sigma$), minimum and maximum length (Min and Max) of generated 150 requests for that particular instance. For all tests throughout the computational results section, we assume that the number of alternatives is equal to six, meaning two services, taxi and shared-taxi with three delayed pickup time slots each.

**Choice simulation.** For each alternative, the systematic part of the utility, Eq. (1) in Section 4.1, is calculated based on the parameters ($\boldsymbol{\beta}$) presented by Krueger et al. (2016). These parameters are presented in Table 4. The in-vehicle time for the taxi service ($I^{taxi}$) is calculated based on the shortest distance traveled with a constant speed of 50 km/h. For shared-taxi, we calculate the in-vehicle time (i.e. maximum ride time) as $\min\{1.5\ I^{taxi}, I^{taxi} + 15\}$. The underlying rationale is to keep shared-taxi as an attractive choice for long-distance trips. The maximum fare of each alternative is then calculated by setting the minimum purchase probability ($\zeta$) to 5% (presented via model $Z$ in Section 4.2). The minimum fare is computed by charging a fixed fare of $ 5 for the service plus the variable charge of $ 0.5/km. The price levels are determined with the discretization level of $ 0.1.

Public transport is considered as the opt-out option (i.e., competition). Here, we assume a cyclic service where on average, customers face 20 min of waiting time (including travel time to/from the nearest station). Moreover, a travel speed of 20 km/h is assumed for this public transport. Each customer has to pay a constant fare of $ 4 for short (less than 5 km) trip and $ 5 for long trips (more than 5 km). To simulate a customer's choice, we first calculate the utility of alternatives being offered. For each alternative, we calculate the systematic part of the utility value based on the above-mentioned procedure. For the stochastic part, we randomly draw value from a standard Gumbel distribution with pre-defined parameters. Among the offered alternatives (including the opt-out option), the one with the highest utility value shows the choice of the customer. Table 5 summarizes all values described above.

### 8.2. Computational performance

In this section, we first present the computational results of assortment selection policies introduced in Section 6.1. Second, the algorithmic performance of dynamic CD-DARP is investigated. Third, we present the computation time-sensitivity related to the routing part. Due to the stochasticity resulted from the customer's choice, we solve each instance five times.

**Assortment selection policies.** Table 6 summarizes the comparison of three assortment selection policies and evaluates them based on the computational time and the overall profit performance. Policy III shows the results related to the complete enumeration of all possible assortments which we use as a benchmark. With Policy III, we ensure that the optimal assortment is offered. The first column, "Disc. Level", represents the price discretization levels. For the price discretization levels of 1 and 0.5, we manage to find the optimal assortment based on Policy III (i.e., complete enumeration) but for price discretization level of 0.1, this is not possible due to high computational time.

The first three rows for the price discretization levels of 1 and 0.5 and the first two rows for level 0.1, we present the computational time associated with each instance. The last column "Ave". indicates the average computation time for each policy. Policy III consistently has the highest calculation time due to the fact that it enumerates all the possible assortments, whereas this value significantly decreases for Policy I and II. In rows entitled "Profit. Change (%)", we compare Policy I and II based on their profit performance for each price discretization level against Policy III (as the benchmark) except for the 0.1 case where we compare Policy I and II against each other. We observe that none of these policies consistently outperforms the others. The reason is that we solve this dynamic problem in a myopic fashion, and we cannot guarantee the global optimality based on the profit performance. On the other hand, the difference between these policies in terms of the profit is not significant. For price discretization levels 1 and 0.1, Policy I outperforms Policy II. Moreover, for most cases, the computational time of Policy I tends to be lower. Therefore, in order to report the rest of our numerical results, we only use Policy I with price discretization level of 0.1.

**Algorithmic performance.** The computational time reported in Table 7 indicates the average time for solving each iteration of CD-DARP. Table 7 is divided into three segments, each one reporting the results associated with a given number of vehicles. For

**Table 6**
Comparison between assortment policies (6 vehicles).

| Disc. | Policy | | Instances | | | | | | | | | | | | Ave. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Level | No. | | 10-US | 10-UM | 10-UL | 10-CS | 10-CM | 10-CL | 15-US | 15-UM | 15-UL | 15-CS | 15-CM | 15-CL | |
| 1 | III | Time | 62.27 | 61.41 | 58.18 | 26.47 | 26.67 | 37.43 | 33.73 | 33.23 | 46.68 | 78.3 | 68.16 | 71.97 | 50.38 |
| | II | Time | 0.05 | 0.05 | 0.06 | 0.06 | 0.08 | 0.08 | 0.03 | 0.04 | 0.03 | 0.05 | 0.06 | 0.05 | 0.05 |
| | I | Time | 0.04 | 0.45 | 0.49 | 0.48 | 0.52 | 0.49 | 0.26 | 0.3 | 0.28 | 0.43 | 0.45 | 0.47 | 0.39 |
| | Profit | (II–III) | −1.23 | −1.41 | −1.2 | −0.56 | −4.25 | 2.56 | −4.03 | −5.41 | −4.28 | 1.60 | 2.20 | −1.82 | −1.49 |
| | Change. (%) | (I–III) | −3.26 | 8.68 | 4.34 | 0.98 | 2.18 | 6.53 | 10.90 | 15.60 | 1.41 | 13.35 | 5.30 | 1.22 | 5.60 |
| 0.5 | III | Time | 459.41 | 700.2 | 1463.07 | 906.26 | 1175.82 | 3607.96 | 2508.59 | 1354.81 | 4943.08 | 4184.25 | 5090.38 | 2849.64 | 2436.96 |
| | II | Time | 2.69 | 3.56 | 4.55 | 3.61 | 3.70 | 4.44 | 2.25 | 2.26 | 2.38 | 2.39 | 3.73 | 5.40 | 3.41 |
| | I | Time | 1.28 | 2.89 | 4.03 | 2.52 | 1.10 | 2.01 | 0.93 | 1.34 | 1.19 | 2.85 | 4.19 | 5.19 | 2.46 |
| | Profit | (II–III) | −0.86 | 0.91 | 8.35 | −0.34 | −2.45 | 10.18 | −1.32 | 2.25 | 15.93 | 0.16 | −5.12 | 0.97 | 2.39 |
| | Change. (%) | (I–III) | −1.69 | 0.98 | −0.5 | −0.78 | −0.94 | 8.23 | −0.43 | 3.16 | 16.05 | −5.68 | −0.18 | −0.47 | 1.48 |
| 0.1 | II | Time | 2.64 | 3.51 | 6.79 | 7.75 | 5.11 | 6.18 | 1.88 | 2.51 | 2.92 | 1.79 | 3.22 | 4.02 | 3.94 |
| | I | Time | 1.6 | 2.79 | 5.5 | 3.9 | 4.32 | 5.54 | 0.31 | 0.35 | 1.83 | 0.94 | 5.01 | 3.47 | 2.96 |
| | Profit Change. (%) | (I–II) | 6.31 | −0.63 | −1.39 | 1.58 | 11.7 | 2.17 | −0.83 | −5.38 | 0.93 | 1.84 | 1.52 | 7.22 | 2.09 |

**Table 7**
The effect of pre-processing steps on CD-DARP computation time (seconds per request).

| No. | | | Instances | | | | | | | | | | | | Ave. | Max |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Vehicles | | | 10-US | 10-UM | 10-UL | 10-CS | 10-CM | 10-CL | 15-US | 15-UM | 15-UL | 15-CS | 15-CM | 15-CL | | |
| Two | No. req. | Ave | 5.32 | 5.74 | 6.6 | 5.1 | 6.69 | 7.54 | 5.38 | 5.93 | 8.14 | 5.54 | 6.1 | 7.38 | 6.29 | 8.14 |
| | | Max | 10 | 10 | 14 | 9 | 12 | 13 | 10 | 11 | 14 | 12 | 11 | 14 | 11.67 | 14 |
| | Base | Ave. | 0.08 | 0.07 | 0.08 | 0.10 | 0.12 | 0.14 | 0.03 | 0.03 | 0.03 | 0.04 | 0.06 | 0.06 | 0.07 | 0.14 |
| | | Worst | 0.93 | 0.50 | 1.05 | 0.82 | 1.05 | 1.06 | 0.30 | 0.35 | 0.08 | 0.45 | 0.40 | 0.39 | 0.61 | 1.06 |
| | +VI | Ave. | 0.04 | 0.04 | 0.05 | 0.05 | 0.06 | 0.07 | 0.02 | 0.02 | 0.02 | 0.04 | 0.05 | 0.04 | 0.04 | 0.07 |
| | | Worst | 0.14 | 0.13 | 0.16 | 0.16 | 0.23 | 0.16 | 0.09 | 0.09 | 0.09 | 0.33 | 0.15 | 0.14 | 0.15 | 0.33 |
| Four | No. req. | Ave | 5.80 | 6.08 | 7.19 | 5.46 | 7.02 | 7.99 | 5.92 | 6.52 | 8.55 | 5.87 | 6.65 | 7.82 | 6.74 | 8.55 |
| | | Max | 10 | 10 | 15 | 9 | 12 | 13 | 11 | 12 | 14 | 12 | 11 | 14 | 11.92 | 15 |
| | Base | Ave. | 0.49 | 0.52 | 0.54 | 0.52 | 1.05 | 1.45 | 0.14 | 0.17 | 0.18 | 0.39 | 0.43 | 0.43 | 0.53 | 1.45 |
| | | Worst | 1.85 | 2.73 | 2.44 | 1.90 | 17.43 | 12.95 | 0.77 | 0.53 | 0.74 | 2.29 | 1.62 | 2.20 | 3.95 | 17.43 |
| | +VI | Ave. | 0.20 | 0.27 | 0.29 | 0.30 | 0.41 | 0.87 | 0.11 | 0.11 | 0.13 | 0.19 | 0.25 | 0.30 | 0.28 | 0.87 |
| | | Worst | 0.52 | 0.77 | 1.14 | 0.84 | 6.39 | 10.32 | 0.36 | 0.39 | 0.39 | 0.47 | 0.82 | 1.46 | 1.99 | 10.32 |
| Six | No. req. | Ave | 5.91 | 6.20 | 7.06 | 5.56 | 7.29 | 8.37 | 5.81 | 6.52 | 8.95 | 6.20 | 6.53 | 7.90 | 6.86 | 8.95 |
| | | Max | 11 | 10 | 14 | 9 | 13 | 14 | 10 | 12 | 15 | 13 | 11 | 14 | 12.17 | 15 |
| | Base | Ave. | 1.54 | 2.08 | 2.59 | 3.17 | 7.50 | 13.68 | 0.44 | 0.56 | 0.64 | 0.99 | 1.93 | 2.18 | 3.11 | 13.68 |
| | | Worst | 10.10 | 10.96 | 30.52 | 18.48 | 85.82 | 97.41 | 1.85 | 2.50 | 3.89 | 3.77 | 28.03 | 15.92 | 25.77 | 97.41 |
| | +VI | Ave. | 1.60 | 2.79 | 5.5 | 3.9 | 4.32 | 5.54 | 0.31 | 0.35 | 1.83 | 0.94 | 5.01 | 3.47 | 2.96 | 5.54 |
| | | Worst | 1.87 | 4.10 | 7.62 | 7.10 | 6.41 | 8.31 | 0.91 | 1.43 | 2.71 | 1.58 | 6.14 | 5.29 | 4.45 | 8.31 |

each arriving request, row 'No. req.' presents the number of existing customers whose routes can still be modified. Similarly, 'Base' reports the computational time after imposing variable reduction steps (explained in Sections 6.2 and 6.3) and tightening constraints with the big Ms (Section 6.4). Row '+VI', shows the results after applying all pre-processing techniques.

As can be seen in Table 7, the computational time of CD-DARP increases by adding the number of vehicles. The worst-case is 97.41 s for a request in instance 10-CL with six vehicles. However, for most of instances the computational time remains low.

On average, adding valid inequalities can slightly reduce the computational time not necessarily for all instances. We however observe that adding valid inequalities are effective in reducing the worst-case computational time when the number of vehicles is increased.

**Computational time sensitivity.** In Table 8, we investigate the impact of varying the number of alternatives on the total computation time. Each element in the table indicates the computational time (in seconds) for every instance and the number of alternatives offered to the customer. Here, we use the first assortment selection policy which results in evaluating $2^{|A|}$ assortments to every arriving customer. We observe that the computational time is sensitive against the number of alternatives. As shown in Table 8 by increasing the number of alternatives the average computational time increases from 2.96 to 31.42 s.

In Table 9, we present the computational sensitivity against the number of available vehicles and inter-arrival times. Earlier, we indicated that, for the sake of instance generations, we assume four and two minutes inter-arrival times between each pair of customers. Here, we discuss the model's limitations when reducing the inter-arrival times, examining four scenarios for tighter inter-arrivals times. We define three additional multipliers ($\frac{1}{2}, \frac{1}{4}, \frac{1}{8}$) to shrink inter-arrival times. We also change the fleet size between 6 and 20, mentioned in the first row of Table 9. We observe a significant jump in the computational time as the fleet size increases and the inter-arrival times are reduced.

**Table 8**
Sensitivity of computational time (in seconds) by varying the number of offered alternatives in the assortment (6 vehicles).

| No. | Instances | | | | | | | | | | | | Ave. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Alt. | 10-US | 10-UM | 10-UL | 10-CS | 10-CM | 10-CL | 15-US | 15-UM | 15-UL | 15-CS | 15-CM | 15-CL | |
| 6 | 1.6 | 2.79 | 5.5 | 3.9 | 4.32 | 5.54 | 0.31 | 0.35 | 1.83 | 0.94 | 5.01 | 3.47 | 2.96 |
| 8 | 15.96 | 26.58 | 31.95 | 28.16 | 33.06 | 35.68 | 5.40 | 11.41 | 12.73 | 14.67 | 27.54 | 30.53 | 22.81 |
| 10 | 32.92 | 34.33 | 35.18 | 34.39 | 35.38 | 35.23 | 21.66 | 23.18 | 25.48 | 31.68 | 33.21 | 34.43 | 31.42 |

**Table 9**
Average CD-DARP computation time (seconds) over all instances.

| Inter-arrival multiplier | No. vehicles | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 |
| 1 | 2.96 | 6.03 | 10.38 | 25.40 | 39.87 | 62.23 | 81.81 | 78.05 |
| $\frac{1}{2}$ | 2.25 | 5.21 | 8.92 | 23.75 | 47.21 | 83.26 | 113.19 | 111.66 |
| $\frac{1}{4}$ | 3.63 | 6.37 | 18.51 | 25.71 | 66.81 | 81.53 | 216.04 | 222.73 |
| $\frac{1}{8}$ | 5.43 | 9.40 | 21.58 | 33.46 | 60.40 | 90.22 | 170.93 | 236.58 |

In our generated instances, we consider the situation in which customers are impatient. To reflect this condition, we assume that customers can wait up to nine minutes and arrive shortly before their preferred pickup time, Rahimi et al. (2020) and Bertsimas et al. (2019). These instances reflect the situation faced by on-demand mobility services. To extend the generality of our proposed framework, we test CD-DARP on the instances introduced by Cordeau (2006). These instances represent situations where customers arrive well in advance and have more flexibility on pickup time (i.e., elderly and handicapped transportation).

The preferred time window is either set at the pickup or drop-off location in a dial-a-ride problem. While in CD-DARP, the preferred time window is set on pickup location. We use the approach proposed by Cordeau (2006) to set a preferred pickup time window for those requests whose drop-off time window is available. The delayed pickup times extend the preferred time window. Moreover, we apply the method proposed by Berbeglia et al. (2012) to make these instances dynamic and compatible with the CD-DARP framework (i.e., no request is known a priori). We consider two situations. For the first situation request arrival is at least 60 min prior to the preferred departure time. For the second situation, this value is increased to 120 min. The rest of the parameters are the same as our previous instances.

In all instances, we set a time limit of 600 s for CD-DARP.

Tables 10 and 11 present the computational results related to the dial-a-ride instances proposed by Cordeau (2006). The first column shows the instance name. For example, "a2-16" shows the instance whose fleet size is 2, serving 16 customers. Columns under "Rejected req". report the number of rejected customers. The first column shows the number of requests rejected by CD-DARP (no feasible assortment), and the second column shows the case in which a customer rejects the assortment offered by CD-DARP. The number of served requests is reported under the column "Rejected req.". The average and the maximum computation time is reported under the column "Time (sec)". We limit the maximum computation time to 600 s per arriving request. Column "Not proved" shows the maximum number of requests in which the solution of CD-DARP reaches its time limit. Column "Active # req." shows the average and the maximum number of requests in which their routes can be re-routed at every iteration of CD-DARP. Finally, the collected revenue and generated cost are reported under the columns "Rev." and "Cost".

As can be seen in Table 10, the computational time slightly increases when the length of the pickup time window is increased due to CD-DARP having more re-routing possibilities. The CD-DARP can be solved within a reasonable amount of time as long as the number of active users is around 11, with 18 active requests in the worst case (e.g., instance a4-32).

However, as the average and the maximum time of active requests increase, the CD-DARP reaches its computational limit of 600 s. For example, for instance, a7-84, CD-DARP cannot be proved to the optimality for ten requests. When request arrival increases to 120 min in advance, the average number of active requests increases from 12.66 to 20.61. This increment increased the computational time of CD-DARP. Still, CD-DARP is solved reasonably when the number of active users is around 11, with 18 in the worst case. By comparing these two situations, we observe that advance arrival information slightly impacts the number of served requests (41.85 compared to 42.34). The average routing cost remain unchanged, and the revenue slightly increases.

**Summary.** The CD-DARP is the combination of assortment optimization and dial-a-ride problems, both of which were proven to be NP-Hard. In Sections 8.1 and 8.2 through an extensive computational experiments we show to what extent CD-DARP can be solved using general optimization solvers. We also show that the computational time is sensitive to the number of offered assortments and the number of active users that will also increase the size of the routing problem. This limitation calls for the introduction of an efficient heuristic approach (similar to the one proposed by Berbeglia et al. (2012)) that can tackle the computational burden. This part has been left out of the scope of this paper and will be tackled in future research.

### *8.3. Flexible time windows*

In dynamic DARP, tight pickup time windows are assumed to provide satisfactory services to customers. However, this could lead to poor system performance based on the number of accommodated customers and the total routing cost. To address this issue,

**Table 10**
CD-DARP performance for instances proposed by Cordeau (2006) - customers arrive 60 min in advance. The reported values are in numbers not percentage.

| Name | Rejected req. | | Served | Time (s) | | Not proved | Active # req. | | Rev. | Cost |
|---|---|---|---|---|---|---|---|---|---|---|
| | Operator | Cust. | req. | Ave. | Max | Max | Ave. | Max | | |
| a2-16 | 3.67 | 1.90 | 10.43 | 0.38 | 0.79 | 0 | 5.50 | 9 | 120.58 | 82.68 |
| a2-20 | 1.48 | 1.95 | 16.57 | 0.13 | 0.26 | 0 | 6.10 | 11 | 174.87 | 126.04 |
| a2-24 | 0.00 | 6.00 | 18.00 | 0.07 | 0.13 | 0 | 5.33 | 8 | 186.19 | 131.39 |
| a3-18 | 0.00 | 1.00 | 17.00 | 0.16 | 0.26 | 0 | 8.30 | 14 | 173.98 | 119.27 |
| a3-24 | 0.00 | 2.00 | 22.00 | 0.17 | 0.36 | 0 | 8.20 | 13 | 210.58 | 134.04 |
| a3-30 | 2.59 | 2.94 | 24.47 | 0.14 | 0.37 | 0 | 8.30 | 13 | 262.20 | 177.93 |
| a3-36 | 1.59 | 0.98 | 33.42 | 0.95 | 6.89 | 0 | 8.90 | 15 | 379.89 | 260.37 |
| a4-16 | 0.00 | 0.03 | 15.97 | 4.74 | 18.57 | 0 | 11.18 | 17 | 169.97 | 119.85 |
| a4-24 | 0.00 | 1.00 | 23.00 | 1.83 | 5.44 | 0 | 10.41 | 15 | 237.28 | 162.33 |
| a4-32 | 0.00 | 3.00 | 29.00 | 9.28 | 47.87 | 0 | 11.18 | 18 | 308.81 | 196.00 |
| a4-40 | 0.00 | 0.10 | 39.90 | 5.43 | 34.11 | 0 | 11.70 | 18 | 408.21 | 268.95 |
| a4-48 | 0.00 | 0.24 | 47.76 | 16.46 | 600 | 1 | 11.30 | 21 | 484.75 | 323.70 |
| a5-40 | 0.00 | 0.30 | 39.70 | 20.50 | 292.54 | 0 | 13.32 | 20 | 394.81 | 237.71 |
| a5-50 | 0.00 | 0.29 | 49.71 | 11.54 | 84.24 | 0 | 13.28 | 19 | 519.11 | 310.73 |
| a5-60 | 0.00 | 1.00 | 59.00 | 15.95 | 188.64 | 0 | 14.35 | 25 | 644.22 | 404.54 |
| a6-48 | 0.00 | 1.00 | 47.00 | 55.94 | 272.16 | 0 | 16.75 | 27 | 497.46 | 285.99 |
| a6-60 | 0.00 | 5.00 | 55.00 | 34.51 | 600 | 2 | 16.08 | 26 | 590.19 | 376.15 |
| a6-72 | 0.00 | 3.00 | 69.00 | 31.67 | 600 | 1 | 16.08 | 23 | 715.26 | 442.94 |
| a7-56 | 0.00 | 1.00 | 55.00 | 54.06 | 600 | 4 | 17.69 | 27 | 571.99 | 333.73 |
| a7-70 | 0.00 | 2.00 | 68.00 | 116.77 | 600 | 4 | 18.55 | 30 | 704.32 | 426.43 |
| a7-84 | 0.00 | 1.00 | 83.00 | 126.07 | 600 | 10 | 19.20 | 29 | 870.07 | 508.92 |
| a8-64 | 0.00 | 0.12 | 63.88 | 97.47 | 600 | 7 | 20.84 | 28 | 638.03 | 353.99 |
| a8-80 | 1.30 | 2.99 | 75.71 | 128.06 | 600 | 7 | 18.56 | 28 | 748.45 | 423.05 |
| Ave. | 0.46 | 1.69 | 41.85 | 31.84 | | | 12.66 | | 435.27 | 269.86 |

**Table 11**
CD-DARP performance for instances proposed by Cordeau (2006) - customers arrive 120 min in advance. Reported values are in numbers and not percentage.

| Name | Rejected req. | | Served | Time (s) | | Not proved | Active # req. | | Rev. | Cost |
|---|---|---|---|---|---|---|---|---|---|---|
| | Operator | Cust. | req. | Ave. | Max | Max | Ave. | Max | | |
| a2-16 | 1.39 | 2.92 | 11.69 | 0.45 | 1.04 | 0 | 8.00 | 14 | 130.17 | 90.64 |
| a2-20 | 0.00 | 3.00 | 17.00 | 0.18 | 0.39 | 0 | 9.40 | 16 | 174.56 | 133.53 |
| a2-24 | 0.00 | 4.00 | 20.00 | 0.11 | 0.18 | 0 | 8.91 | 12 | 210.00 | 149.62 |
| a3-18 | 0.00 | 1.00 | 17.00 | 0.90 | 3.62 | 0 | 13.94 | 21 | 177.73 | 124.41 |
| a3-24 | 0.00 | 2.00 | 22.00 | 0.60 | 2.21 | 0 | 12.25 | 19 | 210.71 | 130.93 |
| a3-30 | 2.83 | 2.91 | 24.26 | 0.50 | 4.09 | 0 | 12.63 | 20 | 269.08 | 180.76 |
| a3-36 | 1.92 | 0.97 | 33.11 | 17.04 | 70.80 | 0 | 14.97 | 25 | 362.39 | 253.77 |
| a4-16 | 0.00 | 1.00 | 15.00 | 87.50 | 195.89 | 0 | 15.06 | 24 | 164.46 | 112.28 |
| a4-24 | 0.00 | 0.13 | 23.87 | 2.41 | 24.18 | 0 | 16.45 | 25 | 249.91 | 164.84 |
| a4-32 | 0.00 | 0.28 | 31.72 | 63.69 | 99.84 | 0 | 19.81 | 32 | 349.97 | 208.54 |
| a4-40 | 0.00 | 0.06 | 39.94 | 22.29 | 249.90 | 0 | 18.65 | 28 | 407.43 | 267.99 |
| a4-48 | 0.00 | 0.17 | 47.83 | 70.55 | 600 | 4 | 18.18 | 32 | 482.29 | 323.01 |
| a5-40 | 0.00 | 0.08 | 39.92 | 90.73 | 600 | 2 | 18.65 | 28 | 395.10 | 230.05 |
| a5-50 | 0.00 | 0.47 | 49.54 | 89.21 | 600 | 3 | 22.92 | 35 | 518.03 | 301.53 |
| a5-60 | 0.00 | 1.00 | 59.00 | 174.23 | 600 | 3 | 23.68 | 35 | 618.04 | 379.38 |
| a6-48 | 0.00 | 1.00 | 47.00 | 294.91 | 600 | 15 | 27.97 | 46 | 478.81 | 281.61 |
| a6-60 | 0.00 | 3.00 | 57.00 | 218.46 | 600 | 17 | 25.68 | 42 | 613.12 | 386.48 |
| a6-72 | 0.00 | 2.00 | 70.00 | 257.83 | 600 | 18 | 27.22 | 38 | 721.55 | 431.82 |
| a7-56 | 0.00 | 0.44 | 55.56 | 316.90 | 600 | 24 | 29.05 | 44 | 576.67 | 326.70 |
| a7-70 | 0.00 | 0.34 | 69.66 | 354.48 | 600 | 23 | 31.37 | 45 | 722.16 | 439.09 |
| a7-84 | 0.00 | 2.00 | 82.00 | 369.16 | 600 | 42 | 31.96 | 47 | 864.08 | 494.57 |
| a8-64 | 0.00 | 1.00 | 63.00 | 451.63 | 600 | 51 | 34.03 | 49 | 632.98 | 351.53 |
| a8-80 | 1.24 | 1.00 | 77.76 | 466.17 | 600 | 55 | 33.17 | 44 | 777.59 | 432.50 |
| Ave. | 0.32 | 1.34 | 42.34 | 145.65 | | | 20.61 | | 439.43 | 269.37 |

in the CD-DARP framework, we assume flexible pickup time windows. In this section, we show how this flexibility can result in better performance of the system.

For a given service type (i.e. taxi, shared-taxi), we have designed three scenarios to assess the benefits of the proposed model by offering customers various pick-up time windows. For this particular experiment, for a fair comparison between scenarios, we assume that only the operator can reject a request due to infeasibility, while customers are captive (no opt-out option). The utility of the opt-out option is still used to determine the price of alternatives. These scenarios are described as follows:

- (**P**). Only the customer's preferred pick-up time is offered. This scenario is the same as the dynamic dial-a-ride problem in which all customers requests contain time window at the pick-up location and the length of the time window is the same for

**Table 12**
Routing cost (%) and service level improvement (%) - taxi service.

| # Veh. | Number of served passengers | | | | | | Total routing cost | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Imp. (%) | 6 | | 4 | | 2 | | 6 | | 4 | | 2 | |
| | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** |
| 10-US | 1.68 | 4.20 | 2.95 | 6.56 | 2.57 | 8.57 | 0.49 | 0.28 | −0.98 | 0.01 | 3.22 | 4.78 |
| 10-UM | 1.64 | 2.74 | 2.05 | 8.66 | 4.66 | 8.22 | −0.85 | −1.66 | 0.35 | 3.43 | −0.82 | 0.60 |
| 10-UL | 1.66 | 3.45 | 2.68 | 6.30 | 2.25 | 7.50 | 0.81 | 1.28 | 0.22 | 2.01 | 1.82 | 4.59 |
| 10-CS | 1.08 | 1.35 | 3.65 | 6.57 | 4.21 | 6.32 | −0.99 | −3.13 | −0.23 | −2.95 | 1.36 | 2.89 |
| 10-CM | 0.00 | 0.00 | 5.52 | 8.96 | 5.32 | 6.38 | −2.00 | −3.94 | 2.82 | 4.63 | 4.13 | 1.57 |
| 10-CL | 1.22 | 1.35 | 3.97 | 8.09 | 7.56 | 17.78 | 1.28 | −1.49 | 2.26 | 2.42 | 3.36 | 6.89 |
| 15-US | 2.48 | 0.83 | 0.00 | −1.06 | −0.39 | 3.92 | 1.54 | −2.85 | −1.86 | −2.50 | −1.65 | −0.92 |
| 15-UM | 0.47 | 3.10 | 4.00 | 6.32 | 2.74 | 1.96 | −2.63 | −0.91 | −1.24 | −2.54 | −1.71 | −5.46 |
| 15-UL | 0.97 | 3.23 | 1.94 | 3.23 | 1.60 | 6.00 | −2.55 | −1.38 | 1.31 | 4.10 | −0.10 | 1.28 |
| 15-CS | 2.10 | 2.10 | 2.17 | 4.17 | 2.77 | 7.69 | 0.36 | −2.91 | −0.11 | 2.58 | 2.13 | 2.33 |
| 15-CM | 1.82 | 4.20 | 4.10 | 9.02 | 3.78 | 5.41 | 1.23 | −0.89 | 3.00 | 7.58 | 4.03 | 7.84 |
| 15-CL | 2.54 | 4.93 | 4.79 | 11.57 | 4.17 | 6.94 | 0.68 | −0.65 | 3.54 | 7.53 | 3.48 | 11.27 |
| Average | 1.47 | 2.62 | 3.15 | 6.53 | 3.44 | 7.22 | −0.22 | −1.52 | 0.76 | 2.19 | 1.61 | 3.14 |

**Table 13**
Routing cost (%) and service level improvement (%) - shared taxi service.

| # Veh. | Number of served passengers | | | | | | Total routing cost | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Imp. (%) | 6 | | 4 | | 2 | | 6 | | 4 | | 2 | |
| | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** | **P → A** | **P → D** |
| 10-US | 0.41 | 1.35 | 2.81 | 6.25 | 4.00 | 14.67 | −4.61 | −9.05 | −3.21 | −3.68 | 1.60 | 4.58 |
| 10-UM | 1.36 | 2.04 | 1.94 | 7.46 | −0.95 | 3.57 | −2.93 | −7.28 | −1.54 | −0.43 | −0.25 | 2.02 |
| 10-UL | 1.77 | 2.04 | 4.03 | 7.46 | 1.15 | 12.64 | −2.74 | −8.21 | −0.64 | −3.47 | −0.64 | 6.05 |
| 10-CS | 0.81 | 1.35 | 2.54 | 2.82 | 7.42 | 11.34 | −3.24 | −9.83 | −2.33 | −10.48 | 1.02 | −2.54 |
| 10-CM | 0.00 | 0.00 | 4.46 | 7.91 | 6.33 | 10.20 | −5.07 | −11.56 | −0.07 | −6.39 | 3.50 | 2.12 |
| 10-CL | 0.00 | 0.00 | 2.52 | 4.90 | 8.00 | 12.63 | −5.09 | −10.19 | −4.67 | −8.16 | 1.94 | 5.16 |
| 15-US | 0.00 | 3.08 | 0.00 | 1.98 | −6.21 | −6.90 | −2.80 | −2.86 | −2.01 | −2.91 | −2.18 | −7.86 |
| 15-UM | 1.34 | 5.22 | 3.62 | 6.67 | 7.55 | 11.32 | −3.94 | −6.12 | −2.60 | −3.10 | −2.98 | −2.23 |
| 15-UL | 2.44 | 8.40 | 3.47 | 10.20 | 3.02 | 5.66 | −3.83 | −5.85 | −2.19 | −0.51 | −0.53 | −0.49 |
| 15-CS | −1.37 | 2.74 | 4.96 | 10.74 | 5.14 | 5.71 | 9.91 | −7.89 | 0.54 | −0.85 | −4.39 | −4.96 |
| 15-CM | 2.62 | 3.45 | 1.07 | 9.16 | 5.71 | 9.09 | −0.57 | −2.78 | −1.70 | 1.33 | 1.15 | −0.36 |
| 15-CL | 1.64 | 2.74 | 5.12 | 8.00 | 6.93 | 14.67 | −4.87 | −10.23 | 0.42 | −2.10 | 1.36 | −2.13 |
| Average | 0.92 | 2.70 | 3.04 | 6.96 | 4.01 | 8.72 | −2.48 | −7.65 | −1.67 | −3.40 | −0.03 | −0.05 |

all of them. In this case, we either accept customers with their preferred pick-up time or we reject them. This case will be used as the benchmark in our computational results. We give the highest flexibility to customers that could jeopardize operational performance.

- (**A**). An assortment is offered to each new customer. The aim is to examine the trade-off between offering a slightly delayed pickup time or offering on-time pickup time. Here, both customers and the operator have a certain level of flexibility.
- (**D**). Only the alternative with the maximum deviation from the preferred pick-up time is offered. In other words, option D offers only the largest time window. With this assumption, we give the highest flexibility to the operator, although that could affect customer satisfaction.

We have considered two indicators separately for each service type, to measure systems' performance: (i) the number of customers served and (ii) total routing cost. Tables 12 and 13 report the improvement of system's performance (in %) across all instances for taxi and shared taxi services, respectively. For different fleet sizes, columns **P → A** report the performance improvement between the dynamic DARP and the case where an assortment of options (i.e., scenario **A**) is offered to each request. Similarly, **P → D** shows the performance improvement when only delayed pickup time options (i.e., scenario **D**) are offered.

For taxi services, by delaying passenger pick-up time (i.e., **P → D**), the number of customers served increases. This improvement is more significant when there is a clear demand-supply mismatch. For example, for our test with two vehicles, there is a 7.22% increase in the number of customers served. Moreover, this improvement is more significant for patient customers, because the system is more flexible in terms of modifying its routes. When comparing the results of Tables 12 and 13, we notice that, in terms of reduced routing cost, when we offer a shared taxi option, we can benefit more significantly from the properties of the proposed CD-DARP compared to the case where only taxis are offered.

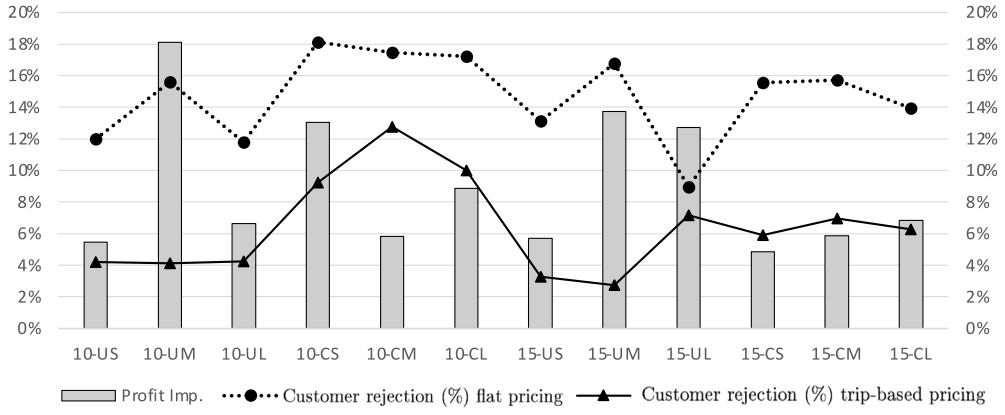



**Fig. 3.** Profit performance (4 vehicles).

### 8.4. Impact of trip-based pricing

In this section, unlike Section 8.3, we assume that customers have the option of rejecting an offered alternative, as well as choosing a more flexible pick-up time window at a lower price. We examine the impact of these assumptions on revenue, service level and operational performance.

We show the impact of trip-based pricing on profit performance for this on-demand system compared to the case where the prices are flat. For the latter case, we consider the average trip length of each instance and we use the same procedure described in Section 3 to calculate the fare.

Fig. 3 compares the profit performance for these two pricing strategies for all instances when the fleet size is four. The dashed and solid lines show the percentage of customers rejecting all offered options for both fixed and dynamic pricing policies. The outcome indicates that even though in both scenarios, unlike dynamic DARP, we offer a set of alternatives, using dynamic pricing improves the service level. The percentage of customers rejecting all offered alternatives reduces on average from 14.7% to 6.41%. Finally, gray columns show profit improvement when dynamic pricing is used (average profit improvement is around 9%).

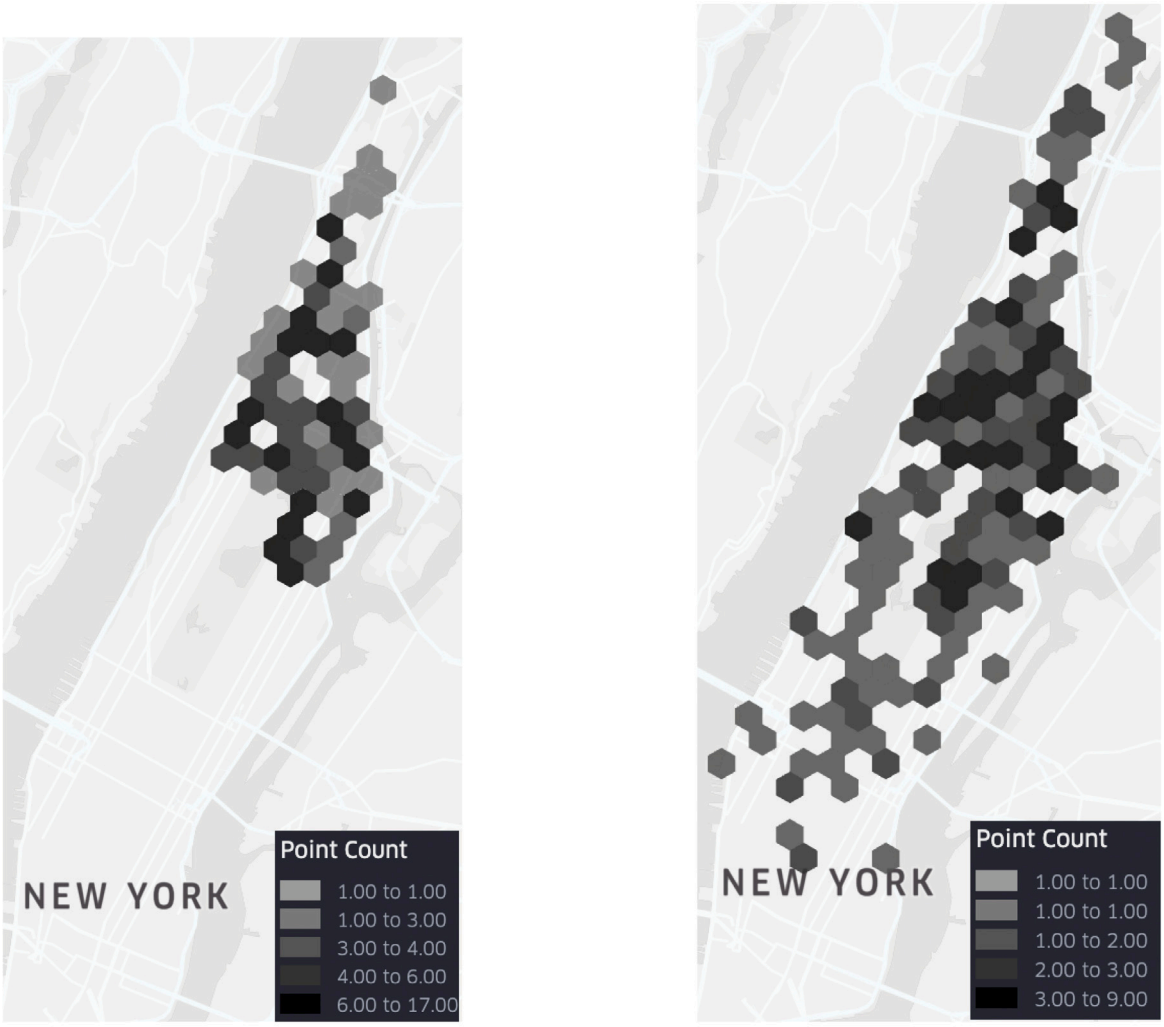
### 8.5. Scaling CD-DARP to real-world data

In this section, we look at the potential of using CD-DARP in practice. We test our proposed algorithm on NYC taxi data, New York City (2016). We use green cab rides of the April 20, 2016 from 9 AM to 10 AM as our sample. We limit our choices to the trips within Manhattan and exclude all trips that either originate or end outside of Manhattan. After removing errors, the data contains 379 observations. Fig. 4 shows the pick-up and drop-off location of our selected case. The chosen case exemplified the situation where most of the rides occurred in a dense area.

From the selected data, we use pick-up time as well as pick-up and drop-off locations. OSRM (Open Source Routing Machine) is applied to compute the travel time and costs, see, Luxen and Vetter (2011) and OpenStreetMap contributors (2017). To adapt the case to our proposed framework, we choose as pick-up time the customer's arrival time. The preferred pick-up time and all values for the choice models, as well as the operating costs are the same as the ones described in Section 8.1. When a new request comes in the system, we solve the CD-DARP to run the experiments with 25 vehicles randomly distributed in the region where we have pick-ups. For each scenario, we solve the model five times and report the average value.

We examine and compare our CD-DARP approach with the current practice, Dias et al. (2017). We define three scenarios to measure the impact of offering assortments to customers and quantify the impact of real-time re-routing on system performance. In scenario I, the algorithm offers customers only the most profitable alternative. Once a request has been assigned to a vehicle, we do not allow for it to change. This scenario mimics the conventional taxi dispatching system. In scenario II, we relax the restriction of offering one alternative and allow for the algorithm to offer an assortment. With this scenario, we aim to evaluate the impact of offering an assortment to the customers. Finally, in scenario III, we offer an assortment and allow the vehicle to be re-routed until the dispatching time. In the last scenario our goal is to identify the opportunities that one can attain by updating vehicle schedule (Section 7.2).

Table 14 denotes the comparison between the three scenarios. Columns under 'Served Cust.' and 'Profit' report the percentage of customers using the service and the total collected profit. As we can see, by moving from scenario I to III, we are able to serve more customers and significantly improve our profit. The detail of offered assortments are presented in columns under 'Assortment %'. The columns 'Taxi', 'STaxi' and 'Mix' denote the share of each service for different scenarios. In the column 'Mix', both the options taxi and shared taxi are offered. Finally, column 'O.Reject' shows the percentages of requests rejected by the operator. When the system offers only one type of alternative, the most profitable one is taxi service. However, in that case, it can only serve 41.68% of the customers. By relaxing this restriction, there is a dramatic change towards offering a mixture of service types (83.83% and 85.17% for scenarios II and III, respectively).




(a) Distribution pick-up locations

(b) Distribution of drop-off locations

**Fig. 4.** Spatial distribution of the pick-up and drop-off locations.

**Table 14**
Operator and customer interaction for three defined scenarios.

| Scn. | Offer | Re | Served | Profit | Assortment (%) | | | | Passengers choice (%) | | | Sat. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Alt. | routing | Cust. (%) | | Taxi | STaxi | Mix | O.Reject | Taxi | STaxi | C.Reject | |
| I | One | No | 41.68 | 436.38 | 96 | 3.5 | 0 | 0.5 | 41.38 | 0.50 | 58.12 | −55.06 |
| II | Menu | No | 74.36 | 654.93 | 0.83 | 5.34 | 83.83 | 10 | 67.03 | 15.59 | 17.38 | −19.99 |
| III | Menu | Yes | 76.36 | 705.13 | 0.67 | 4.33 | 85.17 | 9.83 | 71.36 | 13.32 | 15.32 | −19.38 |

Columns under ‘Passenger choice’, report the result about passenger behavior. From the offered assortments, columns ‘Taxi’, ‘STaxi’ and ‘C.Reject’ show the choice of customers from the offered assortment. As we can see, when we offer the most profitable alternative, passengers are very likely to reject our offer. This is not the case for scenarios II and III. In addition, by allowing vehicle re-routing, we are able to provide a better service to the customer. Finally, column ‘Sat.’ reports customer satisfaction with service being offered. We quantify satisfaction by using the logarithm of the denominator of offered choices (see, Train (2009) for more details). As we can see, by moving from scenario I to III, we can improve customer satisfaction by offering personalized alternatives.

In Section 8.2, we have already acknowledged the computational limitations of the proposed model and its sensitivity against larger fleet size and shorter inter-arrival times. However, the main reason for which we tested our model on the NYC data is to show that by alternative customization (via assortment optimization), the operator is better able to satisfy demand proportionate to the available fleet size within a specific time interval and for a given number of requests.

From a managerial point of view, the outcome is useful for any single owned transport operator with a limited fleet size. Such companies could use the available resources more efficiently by personalizing the services they offer (e.g., a network of autonomous vehicles that offer both private and shared services). The above-mentioned results provide insight into methods and policies that help to find a trade-off between multiple objectives. When an operator offers only the most profitable alternatives, that will affect the customer satisfaction and service levels (assuming the competition is taken into account). In addition, we have observed that introducing flexible time windows (even if they are defined within a tight range) helps to save cost of routing.

## 9. Conclusion

A solution to dynamic DARP is characterized by the presence of three often conflicting objectives: maximizing the number of served customers, minimizing operating costs, and maximizing user convenience (i.e., service quality). Service quality is usually measured in terms of deviations from the desired pick-up time and maximum ride time. In this paper, we introduce an innovative methodological approach called CD-DARP. Using discrete choice theory, we present the service quality as a utility function. Upon arrival of a new customer, we offer an assortment of alternatives to maximize profit. We show the properties of the assortment problem and use it in our model to solve it efficiently. Extensive computational experiments are conducted to highlight the benefits of using our proposed model in practice from an operator and a customer perspective.

In the CD-DARP, no prior knowledge about future demand is assumed. As mentioned in Section 8.2, the computational limitations of this model call for the introduction of an efficient heuristic method to solve larger instances. In this case, we will need to incorporate the knowledge involving future demand into the model. It would also be interesting to develop algorithms that would allow for learning the choice parameters on a continuous basis which would also partly address the issue of perceived impact of the maximum ride time assumptions. Another extension that to consider is to combining the existing framework with rebalancing the empty fleet in the network, to save cost and attract more passengers.

## CRediT authorship contribution statement

**Sh. Sharif Azadeh:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Data curation. **Bilge Atasoy:** Methodology, Writing – original draft, Investigation, Conceptualization, Validation. **Moshe E. Ben-Akiva:** Conceptualization, Methodology. **M. Bierlaire:** Conceptualization, Methodology. **M.Y. Maknoon:** Conceptualization, Writing – review & editing, Software, Visualization, Methodology.

## Acknowledgments

The authors are grateful to Professor Dessouky, the Associate Editor of the journal, as well as both referees for their time to provide us with detailed comments on the earlier version of the manuscript. Their constructive feedback enabled us to significantly improve the quality and presentation of the manuscript.

## References

Atasoy, B., Ikeda, T., Song, X., Ben-Akiva, M.E., 2015. The concept and impact analysis of a flexible mobility on demand system. Transp. Res. C 56, 373–392.
Attanasio, A., Cordeau, J.-F., Ghiani, G., Laporte, G., 2004. Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. Parallel Comput. 30 (3), 377–387.
Beaudry, A., Laporte, G., Melo, T., Nickel, S., 2010. Dynamic transportation of patients in hospitals. OR Spectrum 32 (1), 77–107.
Ben-Akiva, M.E., Lerman, S.R., Lerman, S.R., 1985. Discrete Choice Analysis: Theory and Application to Travel Demand, Vol. 9. MIT Press.
Berbeglia, G., Cordeau, J.-F., Gribkovskaia, I., Laporte, G., 2007. Static pickup and delivery problems: a classification scheme and survey. Top 15 (1), 1–31.
Berbeglia, G., Cordeau, J.-F., Laporte, G., 2010. Dynamic pickup and delivery problems. European J. Oper. Res. 202 (1), 8–15.
Berbeglia, G., Cordeau, J.-F., Laporte, G., 2012. A hybrid tabu search and constraint programming algorithm for the dynamic dial-a-ride problem. INFORMS J. Comput. 24 (3), 343–355.
Berbeglia, G., Pesant, G., Rousseau, L.-M., 2011. Checking the feasibility of dial-a-ride instances using constraint programming. Transp. Sci. 45 (3), 399–412.
Bertsimas, D., Jaillet, P., Martin, S., 2019. Online vehicle routing: The edge of optimization in large-scale applications. Oper. Res. 67 (1), 143–162.
Bongiovanni, C., Kaspi, M., Geroliminis, N., 2019. The electric autonomous dial-a-ride problem. Transp. Res. B 122, 436–456.
Bösch, P.M., Becker, F., Becker, H., Axhausen, K.W., 2018. Cost-based analysis of autonomous mobility services. Transp. Policy 64, 76–91.
Bruck, B.P., Cordeau, J.-F., Iori, M., 2018. A practical time slot management and routing problem for attended home services. Omega 81, 208–219.
Charnes, A., Cooper, W.W., 1962. Programming with linear fractional functionals. Nav. Res. Logist. Q. 9 (3–4), 181–186.
Chen, K.D., Hausman, W.H., 2000. Mathematical properties of the optimal product line selection problem using choice-based conjoint analysis. Manage. Sci. 46 (2), 327–332.
Cordeau, J.-F., 2006. A branch-and-cut algorithm for the dial-a-ride problem. Oper. Res. 54 (3), 573–586.
Cordeau, J.-F., Laporte, G., 2007. The dial-a-ride problem: models and algorithms. Ann. Oper. Res. 153 (1), 29–46.
Coslovich, L., Pesenti, R., Ukovich, W., 2006. A two-phase insertion technique of unexpected customers for a dynamic dial-a-ride problem. European J. Oper. Res. 175 (3), 1605–1615.
Davis, J.M., Gallego, G., Topaloglu, H., 2014. Assortment optimization under variants of the nested logit model. Oper. Res. 62 (2), 250–273.
Dias, F.F., Lavieri, P.S., Garikapati, V.M., Astroza, S., Pendyala, R.M., Bhat, C.R., 2017. A behavioral choice model of the use of car-sharing and ride-sourcing services. Transportation 44 (6), 1307–1323.
Dumas, Y., Desrosiers, J., Soumis, F., 1991. The pickup and delivery problem with time windows. European J. Oper. Res. 54 (1), 7–22.
Häme, L., 2011. An adaptive insertion algorithm for the single-vehicle dial-a-ride problem with narrow time windows. European J. Oper. Res. 209 (1), 11–22.
Ho, S.C., Szeto, W., Kuo, Y.-H., Leung, J.M., Petering, M., Tou, T.W., 2018. A survey of dial-a-ride problems: Literature review and recent developments. Transp. Res. B 111, 395–421.
Horn, M.E., 2002. Fleet scheduling and dispatching for demand-responsive passenger services. Transp. Res. C 10 (1), 35–63.

Huang, D., Gu, Y., Wang, S., Liu, Z., Zhang, W., 2020. A two-phase optimization model for the demand-responsive customized bus network design. Transp. Res. C 111, 1–21.
Hyytiä, E., Penttinen, A., Sulonen, R., 2012. Non-myopic vehicle and route selection in dynamic DARP with travel time and workload objectives. Comput. Oper. Res. 39 (12), 3021–3030.
Jorgensen, R.M., Larsen, J., Bergvinsdottir, K.B., 2007. Solving the dial-a-ride problem using genetic algorithms. J. Oper. Res. Soc. 58 (10), 1321–1331.
Karamanis, R., Angeloudis, P., Sivakumar, A., Stettler, M., 2018. Dynamic pricing in one-sided autonomous ride-sourcing markets. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). IEEE, pp. 3645–3650.
Köhler, C., Ehmke, J.F., Campbell, A.M., 2020. Flexible time window management for attended home deliveries. Omega 91, 102023.
Kök, A.G., Fisher, M.L., Vaidyanathan, R., 2008. Assortment planning: Review of literature and industry practice. In: Retail Supply Chain Management. Springer, pp. 99–153.
Krueger, R., Rashidi, T.H., Rose, J.M., 2016. Preferences for shared autonomous vehicles. Transp. Res. C 69, 343–355.
Lehuédé, F., Masson, R., Parragh, S.N., Péton, O., Tricoire, F., 2014. A multi-criteria large neighbourhood search for the transportation of disabled people. J. Oper. Res. Soc. 65 (7), 983–1000.
Levin, M.W., 2017. Congestion-aware system optimal route choice for shared autonomous vehicles. Transp. Res. C 82, 229–247.
Liu, Y., Bansal, P., Daziano, R., Samaranayake, S., 2019. A framework to integrate mode choice in the design of mobility-on-demand systems. Transp. Res. C 105, 648–665.
Liu, M., Luo, Z., Lim, A., 2015. A branch-and-cut algorithm for a realistic dial-a-ride problem. Transp. Res. B 81, 267–288.
Luxen, D., Vetter, C., 2011. Real-time routing with OpenStreetMap data. In: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. In: GIS '11, ACM, New York, NY, USA, pp. 513–516. http://dx.doi.org/10.1145/2093973.2094062, URL: http://doi.acm.org/10.1145/2093973.2094062.
Mackert, J., 2019. Choice-based dynamic time slot management in attended home delivery. Comput. Ind. Eng. 129, 333–345.
Madsen, O.B., Ravn, H.F., Rygaard, J.M., 1995. A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. Ann. Oper. Res. 60 (1), 193–208.
Marković, N., Nair, R., Schonfeld, P., Miller-Hooks, E., Mohebbi, M., 2015. Optimizing dial-a-ride services in maryland: benefits of computerized routing and scheduling. Transp. Res. C 55, 156–165.
McFadden, D., 1973. Conditional logit analysis of qualitative choice behavior. Front. Econ. 105–142.
Melachrinoudis, E., Ilhan, A.B., Min, H., 2007. A dial-a-ride problem for client transportation in a health-care organization. Comput. Oper. Res. 34 (3), 742–759.
Mitrovic-Minic, S., Laporte, G., et al., 2004. Waiting strategies for the dynamic pickup and delivery problem with time windows. Transp. Res. B 38 (7), 635–655.
Molenbruch, Y., Braekers, K., Caris, A., 2017a. Typology and literature review for dial-a-ride problems. Ann. Oper. Res. 259 (1–2), 295–325.
Molenbruch, Y., Braekers, K., Caris, A., Berghe, G.V., 2017b. Multi-directional local search for a bi-objective dial-a-ride problem in patient transportation. Comput. Oper. Res. 77, 58–71.
New York City, 2016. New york city taxi & limousine commission-trip record data.. URL: https://www1.nyc.gov/site/tlc/about/data.page.
OpenStreetMap contributors, 2017. Planet dump retrieved from https://planet.osm.org. https://www.openstreetmap.org.
Paneque, M.P., Bierlaire, M., Gendron, B., Azadeh, S.S., 2021. Integrating advanced discrete choice models in mixed integer linear optimization. Transp. Res. B 146, 26–49.
Paquette, J., Cordeau, J.-F., Laporte, G., 2009. Quality of service in dial-a-ride operations. Comput. Ind. Eng. 56 (4), 1721–1734.
Paquette, J., Cordeau, J.-F., Laporte, G., Pascoal, M.M., 2013. Combining multicriteria analysis and tabu search for dial-a-ride problems. Transp. Res. B 52, 1–16.
Parragh, S.N., Doerner, K.F., Hartl, R.F., Gandibleux, X., 2009. A heuristic two-phase solution approach for the multi-objective dial-a-ride problem. Netw. Int. J. 54 (4), 227–242.
Parragh, S.N., Pinho de Sousa, J., Almada-Lobo, B., 2015. The dial-a-ride problem with split requests and profits. Transp. Sci. 49 (2), 311–334.
Psaraftis, H.N., 1980. A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. Transp. Sci. 14 (2), 130–154.
Qian, X., Zhang, W., Ukkusuri, S.V., Yang, C., 2017. Optimal assignment and incentive design in the taxi group ride problem. Transp. Res. B 103, 208–226.
Qiu, H., Li, R., Zhao, J., 2018. Dynamic pricing in shared mobility on demand service. arXiv preprint arXiv:1802.03559.
Rahimi, A., Azimi, G., Jin, X., 2020. Examining human attitudes toward shared mobility options and autonomous vehicles. Transp. Res. Part F 72, 133–154.
Rusmevichientong, P., Shen, Z.-J.M., Shmoys, D.B., 2010. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. Oper. Res. 58 (6), 1666–1680.
Santos, D.O., Xavier, E.C., 2015. Taxi and ride sharing: A dynamic dial-a-ride problem with money as an incentive. Expert Syst. Appl. 42 (19), 6728–6737.
Savelsbergh, M.W., 1992. The vehicle routing problem with time windows: Minimizing route duration. ORSA J. Comput. 4 (2), 146–154.
Sayarshad, H.R., Chow, J.Y., 2015. A scalable non-myopic dynamic dial-a-ride and pricing problem. Transp. Res. B 81, 539–554.
Sayarshad, H.R., Gao, H.O., 2018. A scalable non-myopic dynamic dial-a-ride and pricing problem for competitive on-demand mobility systems. Transp. Res. C 91, 192–208.
Schilde, M., Doerner, K.F., Hartl, R.F., 2011. Metaheuristics for the dynamic stochastic dial-a-ride problem with expected return transports. Comput. Oper. Res. 38 (12), 1719–1730.
Schilde, M., Doerner, K.F., Hartl, R.F., 2014. Integrating stochastic time-dependent travel speed in solution methods for the dynamic dial-a-ride problem. European J. Oper. Res. 238 (1), 18–30.
Talluri, K., van Ryzin, G., 2004. Revenue management under a general discrete choice model of consumer behavior. Manage. Sci. 50 (1), 15–33.
Thomas, B.W., 2007. Waiting strategies for anticipating service requests from known customer locations. Transp. Sci. 41 (3), 319–331.
Toth, P., Vigo, D., 1996. Fast local search algorithms for the handicapped persons transportation problem. In: Meta-Heuristics. Springer, pp. 677–690.
Train, K.E., 2009. Discrete Choice Methods with Simulation. Cambridge University Press.
Ulmer, M.W., Thomas, B.W., 2020. Meso-parametric value function approximation for dynamic customer acceptances in delivery routing. European J. Oper. Res. 285 (1), 183–195.
Wang, R., 2012. Capacitated assortment and price optimization under the multinomial logit model. Oper. Res. Lett. 40 (6), 492–497.
Wang, H., Odoni, A., 2016. Approximating the performance of a "last mile" transportation system. Transp. Sci. 50 (2), 659–675.
Xiang, Z., Chu, C., Chen, H., 2008. The study of a dynamic dial-a-ride problem under time-dependent and stochastic environments. European J. Oper. Res. 185 (2), 534–551.
Zhao, X., Yan, X., Yu, A., Van Hentenryck, P., 2018. Modeling stated preference for mobility-on-demand transit: a comparison of machine learning and logit models. arXiv preprint arXiv:1811.01315.